

İSTANBUL TEKNİK ÜNİVERSİTESİ ★ FEN BİLİMLERİ ENSTİTÜSÜ

**AYARLANABİLİR CMOS SINIFLANDIRICI
DEVRELERDE YENİ OLANAKLAR**

**DOKTORA TEZİ
Merih YILDIZ**

Anabilim Dalı : Elektronik ve Haberleşme Mühendisliği

Programı : Elektronik Mühendisliği

NİSAN 2009

**AYARLANABİLİR CMOS SINIFLANDIRICI
DEVRELERDE YENİ OLANAKLAR**

**DOKTORA TEZİ
Merih YILDIZ
(504022107)**

Tezin Enstitüye Verildiği Tarih : 13 Ocak 2009

Tezin Savunulduğu Tarih : 29 Nisan 2009

**Tez Danışmanı : Doç. Dr. Serdar ÖZOĞUZ (İTÜ)
Eş Danışman : Doç. Dr. Shahram MINAEI (DÜ)
Diğer Jüri Üyeleri : Prof. Dr. Cem GÖKNAR (DÜ)
Prof. Dr. Ali ZEKİ (İTÜ)
Doç. Dr. Nil TARIM (İTÜ)
Prof. Dr. Ece Olcay GÜNEŞ (İTÜ)
Prof. Dr. Tülay YILDIRIM (YTÜ)**

NİSAN 2009

ÖNSÖZ

Doktora çalışmalarım boyunca bana fikirler veren ve benden her türlü yardımı esirgemeyen sayın hocalarım Prof. Dr. Cem Gökner, Doç. Dr. Serdar Özoğuz ve Doç. Dr. Shahram Minaei'ye, serim çizimlerinde bana yardımcı olan Engin Deniz'e çok teşekkür ederim. Ayrıca tez izleme sürecindeki öneri ve fikirlerinden dolayı tez izleme komitesinde bulunan Prof. Dr. Ali Zeki ve Doç. Dr. Nil Tarım'a teşekkür ederim. Doktora tezimin başlangıç aşamasında ve devamında desteğinden dolayı da Prof. Dr. M. Sait Türköz hocamı da saygı ile anarım.

Bu çalışmalarım sırasında sabır ve desteği için sevgili eşim Başak Yıldız'a da çok teşekkür ederim.

Nisan 2009

Merih YILDIZ

Y. Mühendis

İÇİNDEKİLER

Sayfa

ÖNSÖZ.....	iii
İÇİNDEKİLER.....	v
KISALTMALAR	vii
ÇİZELGE LİSTESİ.....	ix
ŞEKİL LİSTESİ.....	xi
ÖZET.....	xv
SUMMARY	xvii
1. GİRİŞ	1
1.1 Literatür Taraması	2
1.1.1 Yazılımsal yöntemler.....	2
1.1.2 Donanımsal yapılar	4
1.2 Ele Alınan Konu ve Problem	9
1.3 Çalışmada Yapılanlar	10
2. SINIFLANDIRICI DEVRESİ TASARIMI VE BENZETİMLERİ	13
2.1 Sınıflandırıcı Yapısı Blok Diyagramı.....	13
2.2 Çekirdek Devreler	16
2.2.1 Çekirdek devre-1	16
2.2.2 Çekirdek devre-2	25
2.2.2.1 Çekirdek devre-2'nin zayıf evirtim benzetimleri.....	35
2.2.2.2 Çekirdek devre-2'nin ayırık elemanlar ile gerçekleşmesi	40
2.2.2.3 Çekirdek devre-2'nin serimi	41
2.2.3 Çekirdek devrelerin karşılaştırılması.....	42
2.3 Çekirdek Devreler ile Elde Edilebilen Bölgeler.....	43
3. EĞİK IZGARALI SINIFLANDIRICILAR	45
3.1 Bölgelerin Oluşturulması	45
3.2 ÇAD Devresi ve Benzetimleri.....	50
4. SINIFLANDIRMA ALGORİTMALARI VE DEVRELERE	
UYGULANMASI.....	59
4.1 Fisher Tabanlı Algoritma ile Çift Eşik Doğrularının Bulunmalarına Genel Bakış.....	59
4.1.1 Fisher tabanlı algoritma ile çift eşik doğrularının bulunması.....	60
4.1.2 Fisher tabanlı algoritma ile çift eşik doğrularının bulunmasında genelleştirilmiş hal.....	65
4.2 Eğiticili Perseptron Öğrenme Algoritması ile ÇAD ve ÇD Parametrelerinin Bulunması.....	67
5. SINIFLANDIRICI DEVRE UYGULAMALARI	71
5.1 Kuantalayıcı	71
5.2 Karakter Tanıma.....	72
5.3 İris ve Haberman Verisinin Sınıflandırılması	78
5.3.1 İris verisinin Fisher tabanlı algoritma ile sınıflandırılması	78

5.3.2 Haberman verisinin Fisher tabanlı algoritma ile sınıflandırılması	83
5.3.3 İris verisinin perseptron öğrenme algoritması ile sınıflandırılması	87
5.3.4 Haberman verisinin perseptron öğrenme algoritması ile sınıflandırılması	90
6. SONUÇLAR VE ÖNERİLER.....	95
KAYNAKLAR.....	97
EKLER.....	105

KISALTMALAR

ÇD	: Çekirdek Devre
ÇAD	: Çarpan Devresi
DI	: Dikdörtgen Izgara
YSA	: Yapay Sinir Ağları
KHA	: Kazanan Hepsini Alır
AÇ	: Akım Çoğullayıcı
n-D	: n Boyutlu
CCII	: İkinci Kuşak Akım Taşıyıcı
DO-CCII	: Çift Çıkışlı İkinci Kuşak Akım Taşıyıcı
DBSB	: Dört Boyutlu Sınıflandırıcı Bloğu

ÇİZELGE LİSTESİ

Sayfa

Çizelge 2.1 : Çekirdek Yapı Değişkenleri	16
Çizelge 2.2 : Çekirdek devre-1 yapısının MOS tranzistor boyutları.	20
Çizelge 2.3 : 1×1 -D sınıflandırıcı yapısı çekirdek devre-1 kontrol akımları.....	23
Çizelge 2.4 : 2×1 -D sınıflandırıcı yapısı çekirdek devre-1 kontrol akımları.....	24
Çizelge 2.5 : ÇD-2’de kullanılan MOS tranzistor boyutları.....	30
Çizelge 2.6 : ÇD-2 ile gerçekleştirilen 1×1 -D kontrol akımları (μA olarak).....	34
Çizelge 2.7 : ÇD-2 ile gerçekleştirilen 2×1 -D sınıflandırıcı devresi kontrol akımları (akımlar μA olarak ifade edilmiştir).	34
Çizelge 2.8 : Zayıf evirtimde çalışan ÇD-2 yapısı ile gerçekleştirilen 1-D sınıflandırıcı yapısı kontrol akımları (akımlar nA).	39
Çizelge 2.9 : Zayıf evirtimde çalışan ÇD-2 yapısı ile gerçekleştirilen 2-D sınıflandırıcı yapısı kontrol akımları (akımlar nA).	39
Çizelge 2.10 : Akım kaynakları yerine kullanılan direnç değerleri.....	40
Çizelge 2.11 : Sınıflandırıcı devrelerin karşılaştırılması.	42
Çizelge 3.1 : ÇAD devresi MOS tranzistorların boyutları.	57
Çizelge 5.1 : Kuantalayıcı Yapısında Kullanılan Çekirdek Devre Parametreleri. ...	72
Çizelge 5.2 : Farklı örüntüler için y_i ($i=1, \dots, 5$) çıkış değerleri.....	75
Çizelge 5.3 : n çıkışlı akım çoğullayıcı devresi MOS tranzistorların boyutları.	80
Çizelge 5.4 : Şekil 5.11’deki iris verisi sınıflandırıcısı test kümesi.	81
Çizelge 5.5 : Iris sınıflandırıcısı ÇAD direnç değerleri.....	82
Çizelge 5.6 : Iris sınıflandırıcısı çekirdek devre kontrol akımları.	82
Çizelge 5.7 : Şekil 5.15’deki Haberman verisi sınıflandırıcısı test kümesi.....	86
Çizelge 5.8 : Haberman sınıflandırıcısı ÇAD direnç değerleri.....	86
Çizelge 5.9 : Haberman sınıflandırıcısı çekirdek devre kontrol akımları.....	86
Çizelge 5.10 : Şekil 5.17’deki iris verisi sınıflandırıcısı test kümesi.	88
Çizelge 5.11 : Şekil 5.17’de c_1 sınıfı için ÇAD direnç değerleri ve ÇD akımları.	90
Çizelge 5.12 : Şekil 5.17’de c_2 sınıfı için ÇAD direnç değerleri ve ÇD akımları.	90
Çizelge 5.13 : Şekil 5.19’daki haberman verisi sınıflandırıcısı test kümesi.....	92
Çizelge 5.14 : Şekil 5.19’da c_1 sınıfı için ÇAD direnç değerleri ve ÇD akımları.	93
Çizelge 5.15 : Sınıflandırıcı başarımları karşılaştırması.....	93

ŞEKİL LİSTESİ

Sayfa

Şekil 1.1 : Donanımsal gerçekleşmiş sınıflandırıcı bloğu [36].	5
Şekil 2.1 : Çekirdek yapı geçiş karakteristiği.	13
Şekil 2.2 : Çekirdek yapının blok diyagramı.	13
Şekil 2.3 : $n \times 1$ boyutlu sınıflandırıcı devrenin blok diyagramı.	14
Şekil 2.4 : Tek boyutlu sınıflandırıcı ile elde edilmek istenilen giriş-çıkış karakteristiği.	15
Şekil 2.5 : 2×1 boyutlu çok seviyeli sınıflandırıcı yapısı blok şeması.	15
Şekil 2.6 : MATLAB programı ile elde edilen 2×1 boyutlu çok seviyeli sınıflandırıcı devresinin (x_1-x_2) -y karakteristiği.	16
Şekil 2.7 : Akım-modlu çekirdek devre-1 yapısının blok diyagramı.	17
Şekil 2.8 : Akım-modlu çekirdek devre-1 yapısının geçiş karakteristiği.	17
Şekil 2.9 : Çekirdek devre-1 iç yapısının işlevsel diyagramı.	17
Şekil 2.10 : Çekirdek devre-1'in giriş katı ve evirici.	18
Şekil 2.11 : NOR kapısı ve çıkış katı.	19
Şekil 2.12 : Çekirdek devre-1 yapısı için I_{out} akımının I_{in} akımı ile değişim karakteristiği ($I_{H2} \neq 0$).	20
Şekil 2.13 : Çekirdek devre-1 yapısı için I_{out} akımının I_{in} akımı ile değişim karakteristiği ($I_{H2}=0$).	20
Şekil 2.14 : Çekirdek devre-1 yapısı için V_{out} geriliminin I_{in} akımı ile değişim karakteristiği.	21
Şekil 2.15 : Giriş-çıkış işareti yayılma gecikmesi.	21
Şekil 2.16 : Çekirdek devre-1'in yayılma gecikmesi benzetim sonucu.	22
Şekil 2.17 : Çekirdek devre-1 ile gerçekleştirilen 1×1 -D sınıflandırıcı blok diyagramı.	22
Şekil 2.18 : ÇD-1 ile gerçekleştirilen 1 -D sınıflandırıcı için I_{out} akımının I_{in} akımı ile değişim karakteristiği.	23
Şekil 2.19 : ÇD-1 ile gerçekleştirilen 2×1 -D sınıflandırıcı blok diyagramı.	24
Şekil 2.20 : ÇD-1 ile gerçekleştirilen 2×1 -D sınıflandırıcı ($I_{in1}-I_{in2}$)- I_{out} karakteristiği.	25
Şekil 2.21 : Akım-modlu ÇD-2'nin blok yapısı.	25
Şekil 2.22 : Akım-modlu ÇD-2 blok diyagramı iç yapısı [55].	26
Şekil 2.23 : Eşik devresinin giriş-çıkış karakteristiği.	26
Şekil 2.24 : ÇD-2 yapısının gerçekleştirilme blok şeması.	27
Şekil 2.25 : Pozitif geribeslemeli eşik devresi devre şeması.	28
Şekil 2.26 : Pozitif geribeslemeli eşik devresi osiloskop çıktısı.	28
Şekil 2.27 : Pozitif geribeslemeli eşik devresi benzetim karakteristiği.	28
Şekil 2.28 : Eşik devresi devre şeması.	29
Şekil 2.29 : ÇD-2 sınıflandırıcı devresi giriş-çıkış karakteristiği.	29
Şekil 2.30 : ÇD-2 devre şeması [55].	30
Şekil 2.31 : Eşik devresinin giriş-çıkış karakteristiği.	31
Şekil 2.32 : ÇD-2 yapısı giriş-çıkış karakteristiği.	31
Şekil 2.33 : ÇD-2 yapısı yayılma gecikmesi benzetimi.	32

Şekil 2.34 : ÇD-2 yapısı giriş-çıkış karakteristiği Monte Carlo analizi.....	33
Şekil 2.35 : Tek boyutlu sınıflandırıcının giriş-çıkış karakteristiği.	33
Şekil 2.36 : ÇD-2 ile gerçekleştirilen 1×1-D sınıflandırıcı giriş-çıkış benzetimi.	34
Şekil 2.37 : Zayıf evirtimde çalışan eşik devresi giriş-çıkış karakteristiği.	36
Şekil 2.38 : Zayıf evirtimde çalışan ÇD-2 yapısı giriş-çıkış karakteristiği.	36
Şekil 2.39 : Zayıf evirtimde çalışan ÇD-2 için yayılma gecikmesi benzetimi.	37
Şekil 2.40 : Zayıf evirtimde çalışan ÇD-2 yapısı giriş-çıkış karakteristiği Monte Carlo analizi.	38
Şekil 2.41 : Zayıf evirtimde çalışan ÇD-2 yapısı ile gerçekleştirilen 1-D sınıflandırıcı giriş-çıkış karakteristiği.....	38
Şekil 2.42 : Zayıf evirtimde çalışan ÇD-2 yapısı ile gerçekleştirilen 2-D sınıflandırıcı ($I_{in1}-I_{in2}$)- I_{out} giriş-çıkış karakteristiği.	39
Şekil 2.43 : ÇD-2 yapısının testinde kullanılan akım kaynağı modeli.	40
Şekil 2.44 : ÇD-2 giriş-çıkış karakteristiği osiloskop sonucu.	40
Şekil 2.45 : ÇD-2'nin serim çizimi.	41
Şekil 2.46 : ÇD-2'nin serim sonrası giriş-çıkış karakteristiği.	41
Şekil 2.47 : 2×1 boyutlu sınıflandırıcının farklı çıkış seviyeleri için genel hali.....	44
Şekil 3.1 : Lineer olarak sınıflandırılamayan veri kümesi.....	45
Şekil 3.2 : Eğik Izgaralı veri sınıflandırıcısı.	46
Şekil 3.3 : Dikdörtgen Izgara olmayan veri bölgeleri.....	46
Şekil 3.4 : ÇAD yapısı blok diyagramı sembolik gösterimi.	47
Şekil 3.5 : Şekil 3.2'deki bloğun iç yapısı.	47
Şekil 3.6 : Paralel bağlanmış sınıflandırıcı devresi.....	48
Şekil 3.7 : Eğik ızgaralı veri sınıfları.	48
Şekil 3.8 : Eğik ızgaralı sınıflandırıcı çıkışının 3-D $I_{out}-(x_1-x_2)$ karakteristiği.	49
Şekil 3.9 : Verilerin ÇAD ve ÇD kullanılarak sınıflandırılması.....	49
Şekil 3.10 : ÇAD devresi blok şeması.	50
Şekil 3.11 : CCII devresi blok diyagramı.	51
Şekil 3.12 : CCII devre şeması [65].....	52
Şekil 3.13 : DO-CCII devre blok diyagramı.	52
Şekil 3.14 : DO-CCII devre şeması.	53
Şekil 3.15 : DO-CCII devresi için V_x 'in V_y ile değişim karakteristiği.	53
Şekil 3.16 : DO-CCII devresi için $ \varepsilon_v $ 'nin V_y ile değişim karakteristiği.....	54
Şekil 3.17 : DO-CCII için I_{z+} ve I_{z-} akımlarının I_x ile değişim karakteristiği.	54
Şekil 3.18 : DO-CCII devresi için $ \varepsilon_{i1} $ 'nin I_x ile değişim karakteristiği.....	55
Şekil 3.19 : DO-CCII devresi için $ \varepsilon_{i2} $ 'nin I_x ile değişim karakteristiği.	55
Şekil 3.20 : ÇAD yapısının CCII ve DO-CCII yapıları ile gerçeklemesi.....	55
Şekil 3.21 : ÇAD devresi için I_{z+} ve I_{z-} akımlarının V_y gerilimi ile değişim karakteristiği ($R_2 / R_3 = 5$).	56
Şekil 3.22 : Çeşitli $k=R_2 / R_3$ değerleri için ÇAD devresi $V_y - I_{z+}$ karakteristiği.....	56
Şekil 4.1 : Lineer olarak sınıflandırılamayan örnek veri kümesi.....	59
Şekil 4.2 : Histogram karakteristiği.	60
Şekil 4.3 : Çift eşik doğrularının gösterilimi.	64
Şekil 4.4 : n girişli tek katlı perseptron yapısı.	68
Şekil 5.1 : Sekiz seviyeli kuantalayıcı devre blok yapısı [55].	71
Şekil 5.2 : Sekiz seviyeli kuantalayıcı devresi $I_{in}-I_{out}$ karakteristiği.	72
Şekil 5.3 : Örnek örüntüler.	73
Şekil 5.4 : Şablon hücre gösterimi.	73

Şekil 5.5 : Karakter tanıma sınıflandırıcısı blok diyagramı.....	74
Şekil 5.6 : İlk beş şablon için karakter tanıma sınıflandırıcısı benzetim sonuçları. ..	75
Şekil 5.7 : Son beş şablon için karakter tanıma sınıflandırıcısı benzetim sonuçları..	76
Şekil 5.8 : Hatalı test şablonları.....	77
Şekil 5.9 : Karakter tanıma sınıflandırıcısı hata düzeltme benzetim sonucu.....	77
Şekil 5.10 : İzdüşürülmüş iris verilerinin orijine uzaklıkları.....	79
Şekil 5.11 : İris verisi sınıflandırıcı bloğu (Fisher tabanlı algoritma sonucu oluşturulmuş).....	79
Şekil 5.12 : Akım çoğullayıcı devre şeması [65].....	80
Şekil 5.13 : Şekil 5.11'deki İris verisi sınıflandırıcısı benzetim sonucu.	83
Şekil 5.14 : İzdüşürülmüş haberman verilerinin orijine uzaklıkları.	84
Şekil 5.15 : Haberman verisi sınıflandırıcı bloğu (Fisher tabanlı algoritma sonucu oluşturulmuştur).	85
Şekil 5.16 : Şekil 5.15'deki haberman verisi sınıflandırıcısı benzetim sonucu.....	87
Şekil 5.17 : İris verisi sınıflandırıcı bloğu (öğrenme algoritması sonucu oluşturulmuş).....	89
Şekil 5.18 : Şekil 5.17'deki iris verisi sınıflandırıcısı benzetim sonucu.	90
Şekil 5.19 : Haberman verisi sınıflandırıcı bloğu (perseptron öğrenme algoritması sonucu oluşturulmuş).	91
Şekil 5.20 : Şekil 5.19'daki Haberman verisi sınıflandırıcısı benzetim sonucu.....	93

AYARLANABİLİR CMOS SINIFLANDIRICI DEVRELERDE YENİ OLANAKLAR

ÖZET

Bu çalışmada, ayarlanabilir sınıflandırıcı devreleri ve uygulama alanları incelenmiştir. Sınıflandırma işlemi, benzer özellik taşıyan objelerin farklı özellikte olanlardan ayırt edilmesi şeklinde tanımlanabilir ve otomatik hedef belirleme, yapay zekâ, yapay sinir ağları, analog-sayısal dönüştürücüler, tıbbi tanı, kuantalama, görüntü işleme, istatistik gibi konularda kullanım alanı bulur. Diğer yandan, gerek gerçek dünyada gerekse sayısal dünyada, verilerin sınıflandırılması büyük önem taşımaktadır. Sınıflandırma yöntemleri ilk olarak 1960'lı yıllarda örüntü sınıflandırma adı altında görülmeye başlanmış ve ilişkin algoritmalarda basit yapılar ele alınmıştır; ilk gerçekleştirilen yapıda en yakın komşu yakınsaması kullanılmıştır.

Bugüne kadar sınıflandırma işlemi, çeşitli algoritmalar yardımıyla genellikle yazılımsal olarak yapılmıştır. Oysaki gerçek zamanda çalışma gerektiren bazı uygulamalarda, sınıflandırma işleminin donanımsal olarak da gerçekleştirilmesi önem kazanmaktadır.

Bu amaçla, çalışmanın donanımsal gerçekleştirilmesiyle ilgili kısmında, önce çekirdek devre diye adlandırılan temel bir yapı tasarlanmış ve bu çekirdek devrelerden oluşan çok girişli-çok çıkışlı bir sınıflandırıcı mimarisi geliştirilmiştir. Bu sınıflandırıcı mimarisi ile sınıflandırılabilen ve sınıflandırılamayan veri kümeleri incelenmiş, sınıflandırılamayan veri kümelerinin ayırt edilebilmesi için çekirdek devre yapıları ile kullanılacak Çarpan Devre yapısı gerçekleştirilmiştir. Dolayısıyla gerek sadece çekirdek devre yapıları, gerekse çarpan devre yapıları ile beraber kullanılarak veri kümelerinin uygun kontrol parametreleri ile sınıflandırılabilmesi gösterilmiştir. Bu kontrol parametrelerinin bulunmasını sağlayan eğitim algoritmaları da incelenmiştir.

Sonuç olarak bu çalışmada, sınıflandırma işlemini donanımsal yapılar ile gerçekleştirebilecek, ayarlanabilir, eğitilebilen yeni sınıflandırıcı devreleri tasarlanmış, sağladıkları yeni olanakların gerçek dünyada bulunan ve de önemli olan uygulamalarda incelenmesi ile elde edilen sonuçlar verilmiş ve sınıflandırma konusundaki etkinlikleri ortaya konulmuştur.

NEW POSSIBILITIES IN TUNABLE CMOS CLASSIFIER CIRCUITS

SUMMARY

In this thesis, new possibilities in CMOS classifier circuits and their applications are investigated. The aim of classification is to assign an unknown object to a class containing similar objects. Classifier circuits can find applications in various fields of applied science such as automatic target recognition, real-time object recognition, pattern recognition, artificial intelligence, neural networks, analog digital converters, quantizers and statistics. Therefore, classification is especially important in the real world applications or in the digital world. Firstly, basic classification methods using the nearest neighbourhood algorithm have been seen in 1960 as pattern recognition.

Nowadays classification is generally achieved with the help of some algorithms aided with computer programs. However, hardware implementation of classifier circuits are important for the some applications that require real-time processing.

For that reason, in this thesis firstly hardware implementation of a basic classification unit called core cell is presented. A multiple-input and multiple-output classification topology is constructed with these core cells. The data sets that can be classified or non-classified with that multiple-input and multiple-output classifier circuits have been investigated. A Scaler Circuit has been realized with core cells and used to classify data sets. As a result it is shown that the data sets, with only core cells or together with scaler circuits, can be classified with the appropriate control parameters. Learning algorithms have been investigated, developed and applied to obtain these control parameters.

To conclude, in this thesis custom tunable CMOS classifier circuits have been designed, tested and applied. The test applications have been chosen from real world problems and the results have verified the effective performance of the classifier topologies and circuits.

1. GİRİŞ

Sınıflandırma genellikle otomatik hedef belirleme, yapay zekâ, yapay sinir ağları, analog-sayısal dönüştürücüler, tıbbi tanı, kuantalama, görüntü işleme, istatistik gibi konularda kullanılmaktadır [1, 2]. Bu konuda yapılan çalışmalar yazılımsal ve donanımsal olmak üzere iki kısımda incelenebilir. Literatürde gerek yazılımsal gerekse donanımsal sınıflandırıcılarla ilgili birçok yayınlara rastlanmaktadır [3,4]. Çok yaygın olan yapay sinir ağı tabanlı sınıflandırıcılara ise hem yazılımsal hemde donanımsal olarak bakmak daha doğru olur. Genellikle donanımsal olarak tasarlanmış sınıflandırıcılar, farklı ağ yapıları sentezleyen Yapay Sinir Ağlarının (YSA) gerçeklemesidir. YSA algoritmalarının büyük bir çoğunluğu bilgisayara uyarlanabilmektedir. YSA, biyolojik nöron hücresinin yapısı ve öğrenme özelliklerinden esinlenerek geliştirilmiş bir hesaplama sistemi olup sınıflandırmaya çok elverişlidir. Bu ağların mimarisini ve öğrenme algoritmalarını geliştirmeye yönelik literatürde birçok çalışmalar bulunmaktadır [5, 6]. Bu çalışmaların geliştirdikleri modeller yazılım ve donanım ortamında kullanım sağlamışlardır [7-9]. YSA'nın seçilmesindeki başlıca neden ise, çok karmaşık ve çok fazla parametre içeren durumlarda bile kullanılabilir olmalarıdır [10]. Yazılımsal olarak çalıştırıldıklarında sakıncalarının başında, gerçek zamanda çalıştırılamamaları, öğrenme algoritma süresinin fazla olması, sınıflandırma yapacağı nesnelerin birbirine çok benzer özellikler göstermesi gelir; ayrıca parametre sayısı çok arttığında yavaş çalışır hale gelmeleri ise diğer bir sakıncadır [11]. Bütün bunların ana nedeni, paralel çalışma esasına göre ortaya atılmış YSA'nın, ardışıl makinelere yönelik algoritmalarla çalıştırılmasıdır; dolayısıyla mevcut sınıflandırıcı algoritmalarının yazılımsal olmalarına karşın, hızlı ve gerçek zamanda çalışma arzu edildiğinde, donanımsal olarak gerçeklenmeleri gerekliliği doğmuştur [12].

1.1 Literatür Taraması

1.1.1 Yazılımsal yöntemler

Genel bakış açısından, bir sınıflandırıcı tasarımı iki farklı kategoride ele alınabilir: eğitilmiş ve eğitimsiz tasarım. Eğitilmiş tasarımda daha önce tanımlanmış bilinen bir giriş sınıfının bilinen çıkışlarıyla, sınıflandırıcının aynı girişlere karşı düşen çıkışları karşılaştırılarak bir hata terimi tanımlanır. Bu terim sıfır oluncaya kadar, sınıflandırıcının parametreleri değiştirilir. Eğitimsiz sınıflandırmada ise bir veri eğitim kümesi bulunmayıp, bir yakınlık kuralına göre verilerin gruplandırılmasının yapılması olarak düşünülebilir [2]. Literatürde kullanılan sınıflandırma yaklaşımlarının çoğunluğu eğitilmiş sınıflandırma kategorisinde yer almaktadır [13]. Bu kategoride yapay sinir ağları ve en yakın komşu gibi yaklaşımlar kullanılmaktadır [3]. YSA tabanlı olan sınıflandırma yapıları, günümüzde en yaygın kullanılan, başarımını ispatlamış çok güçlü sınıflandırıcı türleridirler [14]. YSA'ya yönelik ilk çalışma, 1943 yılında Mc-Culloch ve Pitts'in tarafından yapılmıştır [15]. Bu çalışmada basit mantık fonksiyonlarının gerçekleştirildiği matematiksel bir model kullanılmıştır. Daha sonraki çalışmalarda farklı mimari ve öğrenme yapısına sahip ağlar geliştirilmiştir [16]. Bu ağlardan çok katmanlı algılayıcılar ve radyal tabanlı fonksiyon ağları literatürde sınıflandırma gibi pratik uygulamalarda oldukça yaygın olarak kullanılmıştır [17]. Ayrıca YSA'ların genelleme yeteneklerinin olması, çok büyük boyutlu örüntü tanıma uygulamaları açısından da oldukça önemlidir [18, 19].

Sınıflandırma işleminin bir başka yöntemi de, örüntüleri bulundukları uzayda tanımlı bir mesafe fonksiyonuna (metrik) göre, kendilerine en yakın sınıflara minimum hata ile atamaktır [20].

Örüntü sınıflandırma işlemi temel olarak iki gruba ayrılabilir: bunlardan ilki matematiksel ve istatistiksel tabanlı olanlar, ikincisi ise YSA tabanlı algoritmalar. Sınıflandırıcılar başarımlarını açısından karşılaştırıldığında, YSA'ları kullanılarak elde edilen sınıflandırıcı sonuçları istatistiksel tabanlı sınıflandırıcı sonuçlarından daha başarılı olduğu görülmüştür; ancak öğrenme sürelerinin de uzun olduğu bilinmektedir [19-21]. Bazı çalışmalarda karma yöntemler de önerilmiştir [22].

Diğer bir sınıflandırma yöntemi ise destek vektör makinesidir. Bu yöntem iki sınıflı veriye ait nokta kümesini ayıran en iyi bir soyut-düzlem bulmaya çalışan bir tekniktir. Diğer bir ifade ile bu yöntem veri sınıflarını uygun soyut-düzlemler ile ayırmaya çalışır. Uygun soyut-düzlemin bulunması ise sınıflandırılacak veriler ile soyut-düzlemin mesafesinin minimizasyonu ile yapılır. Bu yöntemin sakıncaların başında ise, belirli veriler için en iyi ayırma düzlemi bulunduğundan sonra yeni veriler geldiğinde, sadece yeni verilere değil eski verilere de ihtiyaç duyulması gelir [23].

Literatürde veri sınıflandırması radyal tabanlı fonksiyonlar kullanılarak da yapılmaktadır [24]. Buradaki veri sınıflandırma yönteminde, sınıflandırma sonuçları belli olan belirli veriler için ağ parametreleri saptanır ve bu parametreler benzer dağılım gösteren başka veriler için de kullanılır. Başka bir deyişle tekrar eğitilmesi gerekmemektedir. Ancak radyal tabanlı fonksiyonlar kullanıldığında doğru sonuç elde etme kesinliği azalırken hızlı sonuç alınabilmektedir; oysa destek vektör makinesi ile durum tam tersi olmaktadır [4].

Örüntü sınıflandırma için kullanılan bir yöntem de vektör kuantalayıcıdır. Bu yöntem örüntü tanımda olduğu kadar, özellikle veri sıkıştırılmasında da kullanılmaktadır. Vektör kuantalayıcı yapılarından, giriş dizisinin örnek dizilerden en çok hangisine benzediğine bulmakta yararlanılır [20,25]. Vektör kuantalama, genellikle daha hızlı bir performans elde etmek için, özellikleri birbirine yakın bir işaret kümesinin, tek bir vektörle temsil edilerek, tek bir kodla kodlanması şeklinde tanımlanabilir.

Bu veri sınıflandırma yöntemlerinden başka literatürde, sınıflandırma yazılımlarında kullanılabilen, k-en-yakın komşu (k-th nearest neighbourhood) ve Voronoi algoritması gibi çeşitli yöntemler de vardır [3]. Bu yöntemlerden, Voronoi diyagramı yaklaşımı örüntü sınıflandırmada farklı veri girişlerinin ayırt edilmesi amacıyla kullanılmaktadır [13].

Son olarak, yine literatürde özellikle sınıflandırma işleminin bir alt kümesi olan analog-sayısal dönüştürme işlemi, parça parça lineerleştirme yaklaşımı ve de nöro-bulanık sistemlerde kullanılan trapezoidal geçiş fonksiyonu da ayrıntılı olarak incelenmiştir [26-28]. Trapezoidal geçiş fonksiyonu yardımı ile görüntü işleme konusunda yapılmış çeşitli uygulamalar bulunmaktadır [29].

Ayrıca bir çeşit veri sınıflandırma yöntemi olarak düşünülebilecek kuantalama da, kodlama ve veri sıkıştırma uygulamalarında yaygın olarak kullanılmaktadır [30].

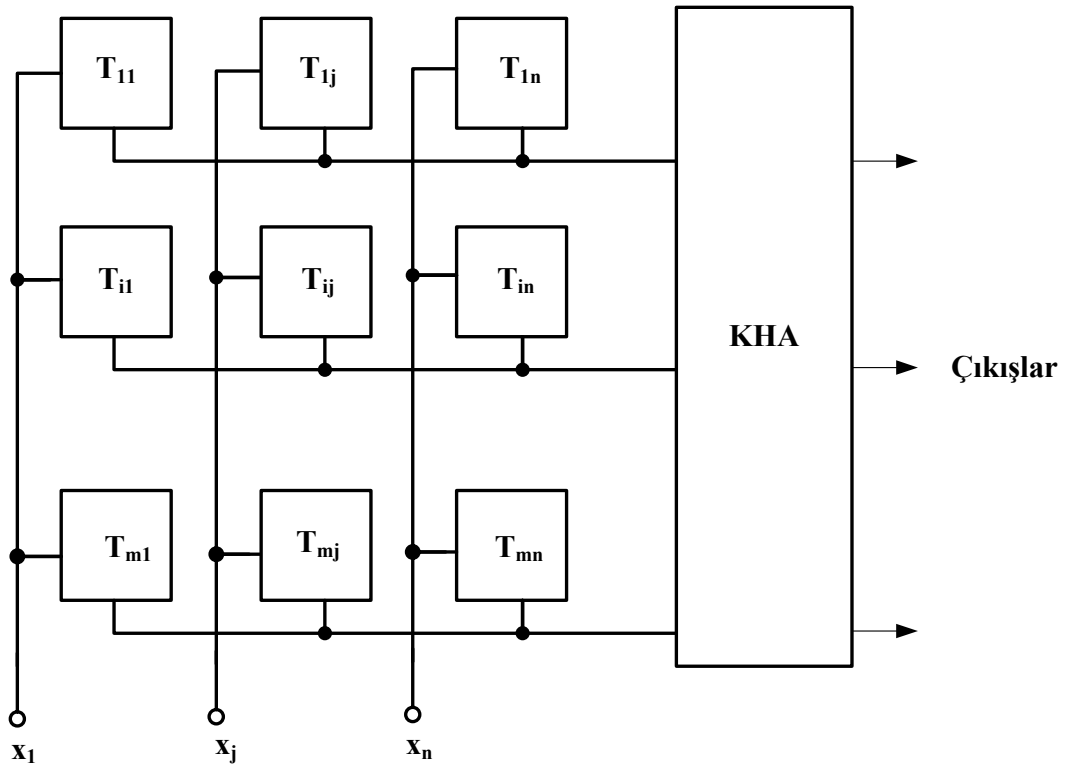
1.1.2 Donanımsal yapılar

Literatürde donanımsal olarak gerçekleştirilmiş sınıflandırıcı yapıları, gerçek zamanlı uygulamalar için elverişli olmaktadır. Bu yapılar genellikle yapı birimi bir nöron modeli olan, farklı ağ yapıları sentezleyen, programlanabilen veya programlanamayan devre ve tümdevre mimarileri olarak karşımıza çıkmaktadır. Özellikle yapay sinir ağlarının donanım gerçeklemeleri olan nöroişlemciler, standart mikroişlemciler ile uygulamalarda kullanılmaktadır. Böylece nörol hesaplanmalardan donanımsal olarak faydalanılmış olunur. Ancak bu donanım uygulamaları giriş-çıkış sayısı, ağ tipi, sabit aktivasyon fonksiyonu, sınırlı çalışma aralığı gibi bazı kısıtlamalar içermektedir. Ancak YSA'nın yazılım ortamında elde edilen esneklik, matematik işlem kabiliyetinin üstünlüğü gibi yeteneklere donanım gerçeklemelerinin getirdiği kısıtlamalar nedeniyle tam olarak ulaşılamamaktadır [5]. Bunun nedeni donanımların yazılım ortamındaki kadar esnek olmayışıdır. Ayrıca yazılım ortamında geliştirilen YSA algoritmalarının donanım uygulamaları için uygun olmayışı, işlem karmaşıklığının yüksek oluşu gibi nedenler gösterilebilir. Literatürde yer alan YSA'nın donanım gerçeklemeleri, uygulama amacına ve kullanılan teknolojiye bağlı olarak çeşitlilik göstermektedir. Bu donanım gerçeklemeleri genel olarak uygulamaya özgü, özel amaçlı tümdevre yapıları ile genel amaçlı tümdevre yapıları olarak karşımıza çıkmaktadır. Belli bir sınıflandırma mimarisi ve algoritmasına uygun tasarlanan tümdevreler dışında standart işlemciler üzerinde de çeşitli topolojiler sentezlenebilmektedir. Uygulamaya özgü tasarlanan tümdevreler, belirli bir uygulamaya göre tasarlandığından, sınırlı büyüklüklere sahip olup böyle bir tümdevrenin başka bir uygulama için kullanımı uygun değildir [31]. Genel amaçlı tümdevreler bu kısıtlamayı en aza indirmek amacıyla tasarlanmıştır. Gerçeklenen tümdevreler genel veya özel amaçlı olması açısından farklılık göstermesi dışında, uygulanan mimarinin analog, sayısal veya karma tasarlanmış olmasına, VLSI tasarım tekniği ve teknolojisine, eğitici veya eğitici olmayan öğrenme kuralına sahip olmasına, ağırlıkların tüm devre üzerinde saklanabiliyor veya saklanamıyor olmasına, tümdevre üzerinde saklanan ağırlıkların analog veya sayısal olmasına, eğitim işleminin tümdevre üzerinde yapılıp yapılmamasına, öğrenme sürecinde kullanılan algoritmaya

bağlı olarak literatürde çeşitlilik göstermektedir [32,33]. Ayrıca literatürde yapay sinir ağı donanımları ile ilgili çalışmalar da ayrıntılı bir şekilde ele alınmıştır [34].

Literatürde kullanılan diğer bir donanımsal sınıflandırıcı yapısı da, yük tabanlı, sabit ağırlıklı olan Hamming sınıflandırıcısıdır [35]. Gerçeklenen bu devrenin üstünlükleri, tek besleme gerilimi (+5 V) kullanılıyor ve statik güç tüketimi olmadan da yüksek hızlı çalışabiliyor olmasıdır; ancak en önemli sakıncası ise ağırlıkları değiştirilemediğinden sabit programlı olmasıdır.

Sınıflandırmada, iki vektör dizisinin farkını alarak sınıflandırma işlemini gerçekleştiren devreler de mevcuttur. Böyle bir devre 0.35 μm CMOS teknolojisi kullanılarak önerilmiştir [36].



Şekil 1.1 : Donanımsal gerçekleştirilmiş sınıflandırıcı bloğu [36].

Donanımsal olarak gerçekleştirilen iki vektör dizisinin farkını hesaplayan genel bir sınıflandırıcı yapısı Şekil 1.1’de gösterilmiştir. Bu şekildeki $(x_1, \dots, x_i, \dots, x_n)$ giriş dizisi vektörünü $((T_{i1}, T_{i2}, \dots, T_{in}), i=1, \dots, m)$ ise giriş vektörüne uzaklıkları hesaplanacak olan sabit örnek vektör dizilerini göstermektedir. Kazanan Hepsini Alır (KHA) bloğu ise hesaplanmış uzaklıkların karşılaştırılıp çıkışa yollandığı bloktur. Gerçeklenen bu devreler, genel olarak iki temel bloktan oluşmaktadır. Uzaklık hesaplamalarının

yapılmakta olduđu ilk bloğun yapısının özelliđi, paralel bađlanmış temel fark alma devrelerinden oluşmasıdır. Bu bloğun yapısı geređi, uzaklık hesabı, algoritma olarak sınıflandırmada kullanılan bir yöntemin donanımsal olarak gerçekleştirilmesiyle yapılmaktadır. İkinci blokta ise genellikle KHA tipinde çıkış katları kullanılmaktadır [37]. Çıkış katlarında kullanılan bu yapıların temel amacı ise basit bir karşılaştırma yapmaktan ibarettir. Bunun nedeni, literatürde önerilen donanımsal sınıflandırıcı yapılarında, iki veri arasındaki Hamming uzaklığının hesaplanması ve elde edilen uzaklık çıktılarının birbirleriyle karşılaştırılması yapılarak sınıflandırma işleminin gerçekleştirilmesidir [38]. Literatürde sabit ağırlıklı Hamming sınıflandırıcılarının kullanılmasının bir başka sebebi ise yüksek hızlı uygulamalara elverişli olmalarıdır. Böyle bir sınıflandırıcı devresi 2.4 μm CMOS teknolojisi kullanılarak gerçekleştirilmiş ve tam sayıların tanınmasında kullanılmıştır; sınıflandırma hızı 10 MHz olarak verilmiştir [39]. Bu tip yapılarda ağırlıkların değişmiyor olması başka uygulamalarda kullanılamaması açısından bir dezavantaj olarak karşımıza çıkmaktadır. Hamming uzaklığı kullanan sınıflandırıcılar analog olarak donanımsal gerçekleştirildiklerinde eşleşme problemi ile hatalı karar vermeye sebep olmaktadır; dolayısıyla eşleşme hataları ve sınırlamaları literatürde incelenmiştir [40].

Sınıflandırma işleminde kullanılan diđer bir yaklaşım ise, yakınlık ölçütü olarak Euclid uzaklığının kullanıldığı yöntemlerdir [41]. Bu yöntemler, hücre devrelerinin paralel bađlanmasıyla temel uzaklık hesabı sağlandığı için, alan verimliliđi açısından çok elverişlidir. Literatürde, temel devre yapısının 4 tranzistor ve 2 kapasite elemanı ile gerçekleştirildiđi örnekler vardır [38]. Ancak temel sakınca, bu devrelerde tasarım esnasında kapasite değerlerinin seçilmesi zorunluluğundan dolayı ağırlık katsayılarının kullanım aşamasında değiştirilememesidir. Dolayısıyla ağırlık katsayılarının değiştirilebilir olması sınıflandırıcıların genel amaçlı kullanımı açısından önem taşımaktadır. Tasarımlarda ağırlık değerleri hem analog hemde sayısal olarak tasarlanmıştır. Bu ağırlık değerlerinin sayısal ve tamsayı biçiminde önerildiđi ve böylelikle kuantalama hatasının oluşmasının da engellendiđi çalışmalar bulunmaktadır [42,43].

Sınıflandırmada başka bir yaklaşımda eğitilici öğrenme algoritmasına dayanan destek vektör makinesidir. Bu destek vektör makinesi doğrusal ayırt edilebilen iki sınıf problemlerinin çözümünden yola çıkarak doğrusal olarak ayırt edilemeyen veya

çoklu sınıf problemlerinin çözümünde kullanılan bir yöntemdir. Gerçek zamanlı sınıflandırma uygulamaları bu öğrenme algoritmasının eğitim süresinin uzun olmasından dolayı zor bir problem olmaktadır. Eğitimde harcanan bu hesaplama süresinin iyileştirilmesi amacıyla kısıtlı bir eğitim kümesi kullanılarak donanımsal olarak destek vektörleri elde edilmiştir. Bu kısıtlı kümeden elde edilen vektörler sınıflandırma başarımını azaltırken, yapının karmaşıklığının da azalmasını sağlamıştır. Literatürde bu şekilde tasarlanmış bir devre 0.5 μm CMOS teknolojisi kullanılarak gerçekleştirilmiş ve güç tüketimi 5.9 mW olarak verilmiştir [44]. Sınıflandırmada destek vektör makinesinin eğitim amacıyla kullanıldığı bir başka çalışmada ise, devre mimarisinde yüzen kapılar kullanılarak, analog işlemci blokları zayıf evirtimde çalışması sonucu nW'lar ile mW'lar mertebesinde düşük güç tüketim değerlerine ulaşılmıştır [45]. Destek vektör makinelerinde radyal tabanlı fonksiyonlar kullanılarak donanımsal olarak gerçekleştirilmiştir [46]. Bu fonksiyonlarda kullanılan Gauss fonksiyonunun ortalama değeri ve varyansı yüzen kapılar kullanılarak ayarlanabilmektedir. Gerçeklemelerin analog tasarımlarında akım aynaları ve logaritma tabanlı süzgeçler de kullanılmıştır. Ancak bu durum geniş yapay sinir ağlarında çok elverişli olmadığı için, direnç ve kuvvetlendiriciler tercih edilmiştir [46]. Ayrıca radyal tabanlı fonksiyon sınıflandırıcısına KHA yapısı da eklenerek analog vektör kuantalayıcıya dönüştürülmüştür [47].

Karar ağaçları da sınıflandırmada kullanılan yöntemler arasında yer almaktadır. Bu yöntem geniş veri kümelerinde dahi yüksek doğrulukla çalışmaktadır. Ancak veri boyutlarının geniş olduğu uygulamalarda yoğun algoritmalarından dolayı işlem süresi çok uzun olmaktadır. Bunun için karar ağacı sınıflandırıcısı donanımsal olarak FPGA (Field Programmable Gate Array) yapıları kullanılarak gerçekleştirilmiştir [48]. Sayısal tasarımların gürültü bağımsızlığının iyi olmasından dolayı, yüksek doğruluğa sahip çıkışlar üretebilmektedir. Ayrıca sayısal tasarım tekniklerinden dolayı, donanım tanımlama dilleri (VHDL) kullanılarak gerçekleştirilen sayısal bir tasarım, FPGA yongalarına kolayca aktarılabilen ve hızlı prototipler üretilmektedir. Donanım tanımlama dilleri kullanarak tasarlanmış sınıflandırıcı mimarileri literatürde bulunmaktadır [49]. Sayısal tasarımların gürültü bağımsızlığı ve tasarım kolaylığı gibi avantajlarının yanında, sayısal yapı blokları ile fonksiyon gerçekleştirmek analog devrelere göre çok daha fazla sayıda tranzistor gerektirmektedir. Yapılan

alıřmalarda genelde analog ve sayısal tasarım tekniklerinin olumlu zelliklerini bir arada kullanan karma tmdevre tasarımları yer almaktadır.

Btn bu yukarıda sz edilen sınıflandırıcı donanımlarına ek olarak literatrde rnt tanımda kullanılan donanımsal sınıflandırıcı yapısı da karřımıza çıkmaktadır. Donanımsal olarak, 2 μm CMOS teknolojisi ile $2.2 \times 2.2 \text{ mm}^2$ alan zerine gereklenmiř ve 6 adet nron ieren rnt sınıflandırıcı yapısı mevcuttur [50]. Bu yapıda kullanılan ğrenme kısmı mikrobilgisayar aracılığı ile gerekleřtirilmiřtir ve devre mW'lar mertebesinde g tk etmektedir [50]. Yz ve karakter gibi karmařık grsel rntlerin sınıflandırılması iin resimlerin sayısal bir kamera yardımıyla alınması ve sayısal iřaret iřleme teknikleri kullanılarak mikroiřlemciler aracılığı ile iřlenmesi saėlanır. Ancak bu iřlemler byk miktarda g tketimi gerektirmektedir. Bunların daha az g tk ederek gerekleřtirilmesi iin analog donanımsal grsel rnt sınıflandırıcılar nerilmiřtir [51]. Literatrde +5 V besleme gerilimi ile beslenen, 0.35 μm CMOS teknolojisi ile gereklenen ve 1.25 mW g tk eden uygulamalar bulunmaktadır [51].

rnt tanımda birok sınıflandırıcıyı birleřtirerek kullanmak geliřmiř bir yntem olarak bilinmektedir. Literatrde karar aėalarının sınıflandırıcı olarak kullanıldığı rnt tanıma devreleri de mevcuttur [52]. Ancak bu tip sınıflandırıcılarda genel bařarım yksek dahi olsa fazla miktarda bellek gereksinimi ve hesaplama sresinin uzunluėu bir dezavantaj olmakta ve gerek zamanlı kullanılmalarına engel teřkil etmektedir. Akıllı fotosensrlerde, dřk znrlkl rnt sınıflandırıcı uygulamalarında kullanılabilirler. Ayrıca yapay sinir aėlarının farklı rnekler iin eėitilebilme yetenekleri sayesinde fotosensensr dizileri ile birlikte kullanılarak CMOS devreler ile gereklenmiř rnt sınıflandırma uygulamaları mevcuttur [53] .

Bylece, yukarıda literatrdeki mevcut sınıflandırıcı devreleri incelenmiř ve bu sınıflandırıcıların donanımsal olarak gerekleřtirilmesi iin pek ok farklı yapıda donanıma zel ğrenme algoritmaları geliřtirilmiřtir. Donanımsal sınıflandırıcı tasarım teknikleri analog, sayısal ve karma olacak biimde tmdevre yapıları gereklenmiřtir. Bu yapıların bir kısmı genel amalı, bir kısmı da zel amalı tasarımlardır.

Tüm bu tasarımlarda, temel sınıflandırıcı yapısı kırmık alanında fazla yer kaplamayacak, güç tüketimi düşük olacak ve üretilmiş olan temel sınıflandırıcı tümdevreleri aralarında bağlanabilecek ve böylelikle çok girişli devrelerin elde edilmesine olanak tanıyacak bir sınıflandırıcı devresi gerçekleştirilmelidir. Ayrıca bu tümdevre gerçeklemesi ağırlıkları dışarıdan ayarlanabilecek şekilde tasarlanmalı ve ağırlık katsayılarının bulunması için tümdevreye uygun öğrenme algoritmaları geliştirilmelidir.

1.2 Ele Alınan Konu ve Problem

Bu çalışma kapsamında ele alınacak sorunlar ve beklentiler aşağıdaki gibi özetlenebilir.

- a) Literatürde, genel amaçlı esnek bir şekilde kullanılabilecek gerek kırmık alanında az yer kaplayacak, gerekse düşük güç tüketimine sahip olacak ayarlanabilir temel bir sınıflandırıcı yapısına ihtiyaç vardır. Sözü edilen sınıflandırıcı temel devrelerin aynı şekilde paralel, seri, ard arda ya da herhangi bir biçimde bağlanarak daha yüksek boyutlu verilerin sınıflandırılmasına olanak vermesini sağlayacak sınıflandırıcı yapılarına gereksinim duyulmaktadır.
- b) Diğer taraftan literatürde incelenen devrelerin çok önemli bir sakıncası da sabit programlı olmalarıdır. Başka bir deyişle bu devrelerin çeşitli veri tiplerine göre, verileri sınıflara ayıran bölgeleri belirleyen devre parametrelerinin kullanıcı tarafından saptanması ve/veya bir uygulamadan ötekine değiştirilebilmesine imkan tanıyacak tasarımlar gerçekleştirilmelidir.
- c) Yukarıda a) şıkında bahsedilen devreler ile çok özel bir yapıda olan dikdörtgen ızgara bölgelerle ayrıştırılabilen veri tipleri sınıflandırılabilir. Bu veri tipleri dışında kalan verilerin sınıflandırılması için uygun bir yöntem geliştirilmelidir. Ayrıca bu tasarımlara uyarlanabilecek öğrenme algoritmaları da oluşturulmalı ve yapılara uyarlanmalıdır.

1.3 Çalışmada Yapılanlar

Tez çalışmasının sınıflandırıcı devrelerinin tasarlanmasında yürütülen yöntem aşağıdaki şekilde özetlenmiştir:

Bölüm 2’de sınıflandırma amacıyla, önce 1.2 a)’da sözü edilen iki adet temel sınıflandırıcı devre bloğu tasarlanmıştır. Tasarlanan sınıflandırıcı devre blokları temel bir yapı niteliği taşımakta; diğer bir ifade ile tek boyutlu bir sınıflandırıcı devresi gibi davranmaktadırlar. Bu yapıya çekirdek devre denilmiş, hem yazılımsal hem de donanımsal gerçekleştirilerek benzetimleri ve karşılaştırılmaları sunulmuştur. Bu devreler ile elde edilebilecek sınıflandırma bölgeleri incelenmiş, bu devreler ile kurulan değişik topolojiler sayesinde ne tür farklı veri tiplerinin sınıflandırılabilirdiği gösterilmiştir. Çekirdek devrelerin çeşitli şekilde bağlanması ile çok boyutlu sınıflandırıcı devreleri gerçekleştirilmiştir. Böylece sınıflandırıcı devrelerin bazı veri tipleri için esnek yapıda tasarlanmış olmasının getirdiği kullanım kolaylıkları da gösterilmiştir. 1.2 b)’de, çekirdek devrenin kontrol parametrelerinin kullanıcı tarafından saptanabilmesi sağlanmıştır. Bu bölümde ayrıca devre parametrelerindeki değişimin devre davranışına etkisini incelemek amacıyla Monte Carlo analizleri yapılmış ve değişim etkisinden söz edilmiştir.

Yukarıda 1.2 c) ile ifade edilen sorunu gidermek amacıyla çekirdek devrelerin önüne yeni bir kat eklenerek oluşturulan Çarpan Devresi (ÇAD) ve benzetimleri Bölüm 3’te incelenmiştir. ÇAD yapısının nasıl gerçekleştirildiği ve bu ÇAD yapısı ile önerilen devrelerin beraber kullanımı ile hangi tür verilerin sınıflandırılabilirdiği de gösterilmiştir.

Önceki bölümlerde önerilen yapılar, özgün ve farklı olduklarından sınıflandırma algoritmaları ve geliştirilen devrelere uygulanması Bölüm 4’te ele alınmıştır. Fisher tabanlı algoritma, perseptron öğrenme algoritmaları incelenmiş, bu algoritmaların önerilen devreler ile nasıl kullanılacağı gösterilmiştir. Çift eşik doğrularının bulunmasında Fisher’in lineer diskriminant analizi yönteminden yararlanılarak yeni bir yöntem geliştirilmiştir.

Gerçekleştirilmiş olan sınıflandırıcının uygulamalarına Bölüm 5’de yer verilmiştir. Kuantalayıcı ve karakter tanıma uygulaması, önerilen devrelerin çeşitli uygulama

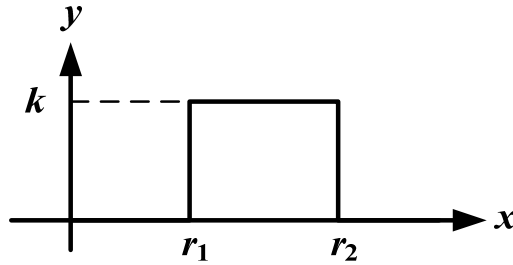
alanlarında kullanılmasına örnek olarak verilmiştir. Geliştirilen sınıflayıcının veri kümelerine uygulanması amacıyla İris ve Haberman verileri Fisher tabanlı ve perseptron öğrenme algoritmaları ile sınıflandırılmış, sonuçlar karşılaştırılarak üstünlükleri gösterilmiştir.

6. bölüm tez kapsamında yapılan çalışmalar ve elde edilen gelişmelere ilişkin sonuç ve yorumların verilmesine ayrılmıştır. Bu bölümde ayrıca, tezde ele alınmış olan problemin devamı niteliğinde yapılabilecek çalışmalar için öneriler de bulunulmuştur.

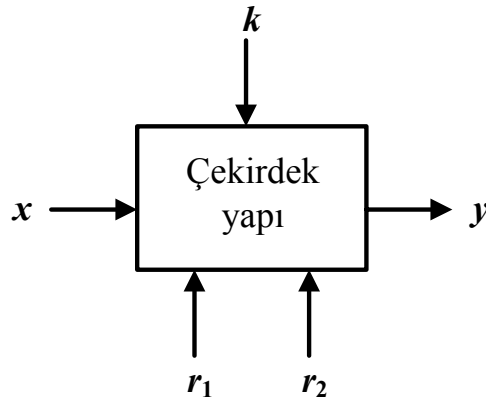
2. SINIFLANDIRICI DEVRESİ TASARIMI VE BENZETİMLERİ

2.1 Sınıflandırıcı Yapısı Blok Diyagramı

Çalışmada öncelikle, basit donanımlarla sınıflandırıcıların nasıl gerçekleştirilebileceği ele alınmış ve sınıflandırma işlemi için temel bir çekirdek yapı ve bu yapının sınıflandırabileceği veri kümeleri incelenmiştir. Bu temel çekirdek yapılar ile oluşturulan topolojilerle daha karmaşık veri kümelerinin de sınıflandırılabilceği gösterilmiş ve gerçeklemeleri yapılmıştır. Çekirdek yapı (ÇY) taşı tek boyutlu bir sınıflandırıcı yapısı olarak düşünülmüş ve tasarlanmıştır. Gerçeklenmesi istenen geçiş eğrisi Şekil 2.1 ve blok diyagramı da Şekil 2.2’de verilmiştir. Bu çekirdek yapı temel sınıflandırıcı yapısını oluşturmaktadır ve çok boyutlu sınıflandırıcı yapılarının elde edilmesinde kullanılmıştır.



Şekil 2.1 : Çekirdek yapı geçiş karakteristiği.

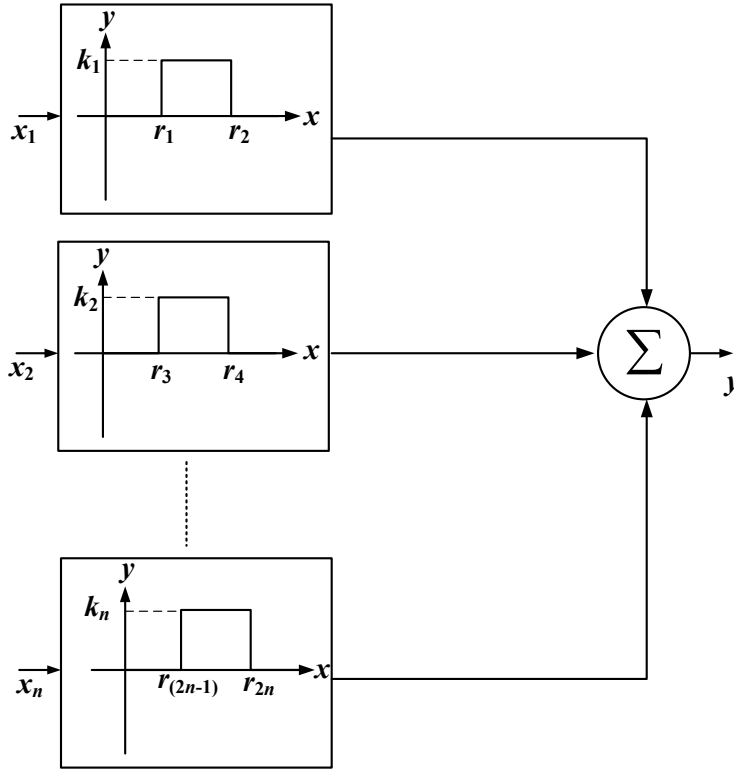


Şekil 2.2 : Çekirdek yapının blok diyagramı.

Tasarımım, Şekil 2.1'deki karakteristikteki r_1 , r_2 ve k parametreleri dışarıdan kullanıcı tarafından kolayca değiştirilebilecek ve uygulanacakları sınıflama probleminden kolayca elde edilebilecek biçimde yapılmıştır. Giriş-çıkış karakteristiği aşağıdaki gibi ifade edilir:

$$y = \begin{cases} k & r_1 < x < r_2 \\ 0 & \text{diğer hallerde} \end{cases} \quad (2.1)$$

Bu yaklaşımın bir üstünlüğü de çekirdek yapıların çeşitli şekillerde aralarında bağlanarak farklı tipteki ve boyuttaki verileri sınıflandırmak için kullanılabilmesidir; örneğin Şekil 2.3'de gösterildiği gibi paralel bağlanarak $n \times 1$ boyutlu bir sınıflandırıcı sistemi gerçekleştirilebilir. Daha karmaşık topolojiler ileriki aşamalarda verilecektir.



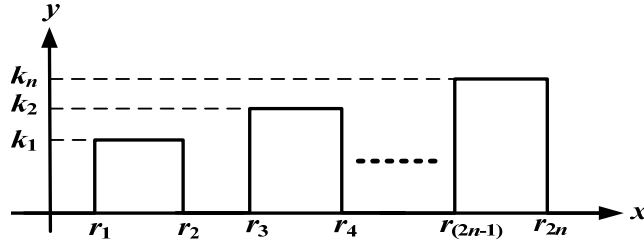
Şekil 2.3 : $n \times 1$ boyutlu sınıflandırıcı devrenin blok diyagramı.

Şekil 2.3'deki yapının çeşitli özel halleri göz önüne alınabilir; bir özel durum

$$x_1 = x_2 = \dots = x_n = x, \quad r_1 < r_2 < r_3 < \dots < r_{(2n-1)} < r_{2n} \quad (2.2)$$

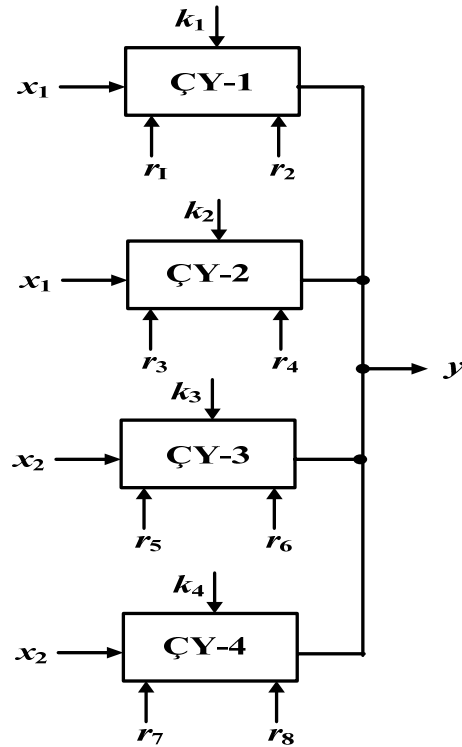
şeklinde verilebilir.

Bu durum için Şekil 2.4’de gösterilen giriş-çıkış karakteristiği elde edilir. Şekil 2.4’deki yapıdan da görüldüğü gibi bu şekilde gerçekleşmiş bir topoloji, tek boyutlu verileri, çıkış büyüklüğünün genliğine bakarak n tane farklı sınıfa ayırabilmektedir. Bu yapıdan aynı zamanda x büyüklüğünün kuantalanmasında da yararlanılabilir.



Şekil 2.4 : Tek boyutlu sınıflandırıcı ile elde edilmek istenilen giriş-çıkış karakteristiği.

Sınıflandırıcı yapılarının diğer bir uygulaması olarak, iki boyutlu ve çok seviyeli bir sınıflandırıcı tasarımı Şekil 2.5’deki topoloji ile elde edilebilir. Bu şekilde verilen topolojide çekirdek yapı-I ve II’ye x_1 giriş değişkeni, çekirdek yapı-III ve IV’e x_2 giriş değişkeni uygulanmıştır. Tasarlanan devreler akım modunda çalıştıklarından çıkışta bir toplama devresi yoktur. Böylece elde edilen sınıflandırıcının giriş-çıkış karakteristiğini daha iyi görebilmek amacıyla kontrol değişkenlerinin Çizelge 2.1’de verilen değerleri kullanılarak MATLAB benzetimi yapılmıştır.

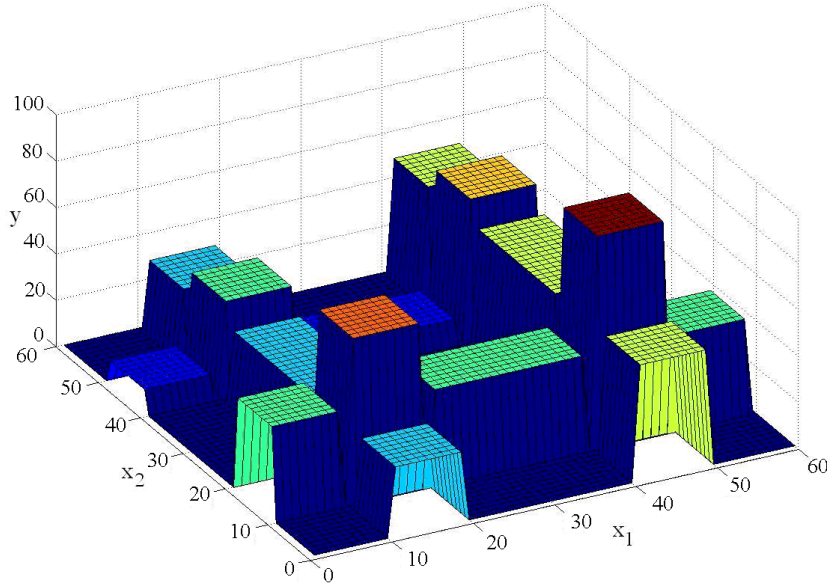


Şekil 2.5 : 2x1 boyutlu çok seviyeli sınıflandırıcı yapısı blok şeması.

Çizelge 2.1 : Çekirdek Yapı Değişkenleri

Çekirdek Yapı											
I			II			III			IV		
r_1	r_2	k_1	r_3	r_4	k_2	r_5	r_6	k_3	r_7	r_8	k_4
10	20	10	40	50	40	10	20	30	40	50	50

Bu durumda Şekil 2.6'daki gibi bir giriş-çıkış karakteristiği elde edilebilir. Bu karakteristikten görüldüğü üzere 2 girişli, 1 çıkışlı ve çok seviyeli (bu örnekte 8 farklı sınıf, 0 düzlemi de dahil edilirse 9 farklı sınıf) sınıflandırıcı yapısı elde edilir.



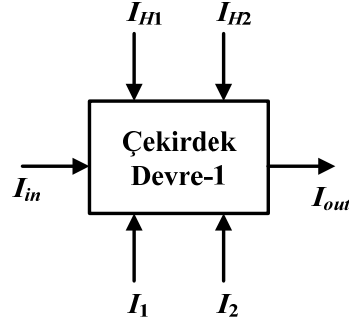
Şekil 2.6 : MATLAB programı ile elde edilen 2×1 boyutlu çok seviyeli sınıflandırıcı devresinin (x_1-x_2) -y karakteristiği.

Tasarlanmış ve matematiksel modeli verilmiş devrenin gerçekleştirilmesi Bölüm 2.2'de ele alınmıştır.

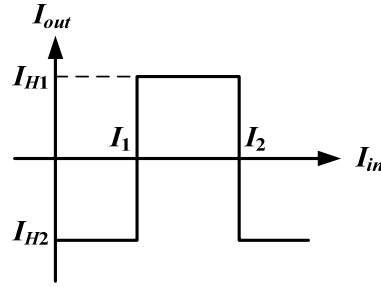
2.2 Çekirdek Devreler

2.2.1 Çekirdek devre-1

Çalışmada tek boyutlu bir sınıflandırıcı devresi olarak, akım modunda çalışmak üzere tasarlanan ve çekirdek devre-1 olarak adlandırılan yapının işlevsel davranışını gösteren blok diyagramı ve geçiş karakteristiği sırasıyla Şekil 2.7 ve Şekil 2.8'de verilmiştir [54]. Şekil 2.8'deki geçiş karakteristiğinden de görüldüğü gibi giriş değerleri I_1 ve I_2 arasında ise çıkış I_{H1} değerini, bunların dışında ise çıkış I_{H2} değerini almaktadır.

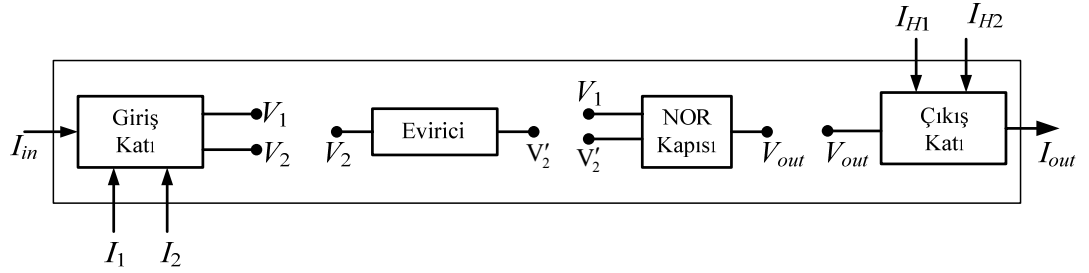


Şekil 2.7 : Akım-modlu çekirdek devre-1 yapısının blok diyagramı.



Şekil 2.8 : Akım-modlu çekirdek devre-1 yapısının geçiş karakteristiği.

Tasarlanmış olan çekirdek devre-1'in işlevsel blok diyagramı Şekil 2.9'da verildiği gibidir. Bu blok diyagramında giriş katı, giriş akımını gerilime dönüştürmek için kullanılırken NOR kapısı ile çıkışın, I_1 ile I_2 akımların arasında I_{H1} değerini alması sağlanmıştır. Karakteristiğinde sıfır olmayan kısmın elde edilmesi için kullanılmıştır. Çıkış değişkenin akım olması ise çıkış katı ile sağlanmaktadır.



Şekil 2.9 : Çekirdek devre-1 iç yapısının işlevsel diyagramı [54].

Şekil 2.8'de verilen giriş-çıkış karakteristiğinin tanım bağıntısı aşağıda gösterildiği biçimdedir:

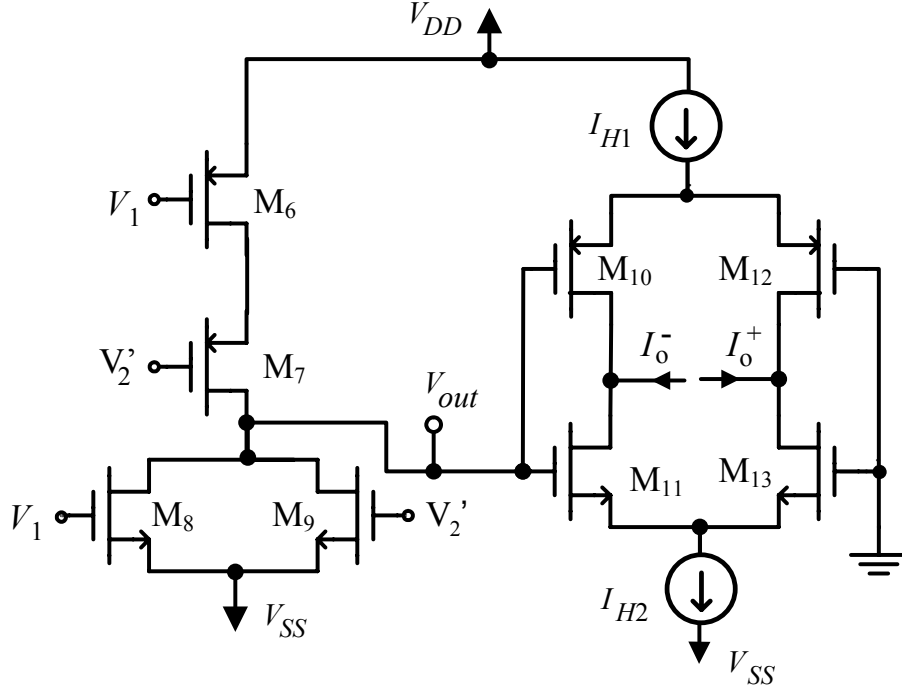
$$I_{out} = \begin{cases} I_{H1} & I_1 < I_{in} < I_2 \\ -I_{H2} & \text{diğer hallerde} \end{cases} \quad (2.3)$$

[illegible]

Buradaki diyot bağılı M_1 tranzistörü ve I_{BIAS} akımı giriş akımını gerilime dönüştürmek için kullanılır. M_2 ve M_3 tranzistörleri de $I_{BIAS}+I_1$ ve $I_{BIAS}+I_2$ akımları ile kutuplanmıştır. Dolayısıyla devrede kullanılacak olan iki farklı eşik değeri elde edilmiş olur. Diğer bir deyişle buradaki M_1 , M_2 ve M_3 tranzistörlerinden oluşan kat, hem bir eşik devresi hem de akımı gerilime dönüştürmek amacıyla kullanılmıştır. M_4 ve M_5 tranzistörleri ise evirici olarak çalışmaktadır. Giriş katı ile V_1 ve V_2' gerilimleri elde edilmiştir.

$$V_{out} = \begin{cases} V_{DD} & I_1 < I_{in} < I_2 \\ V_{SS} & \text{diğer hallerde} \end{cases} \quad (2.4)$$

Son olarak, kullanılan M_{10} , M_{11} , M_{12} ve M_{13} tranzistörleri çıkış katını oluşturmaktadır. Çıkış katında kullanılan I_{H1} ve I_{H2} akım kaynakları geçiş karakteristiğindeki tepe noktalarının ayarlanmasını sağlamaktadır.



Şekil 2.11 : NOR kapısı ve çıkış katı.

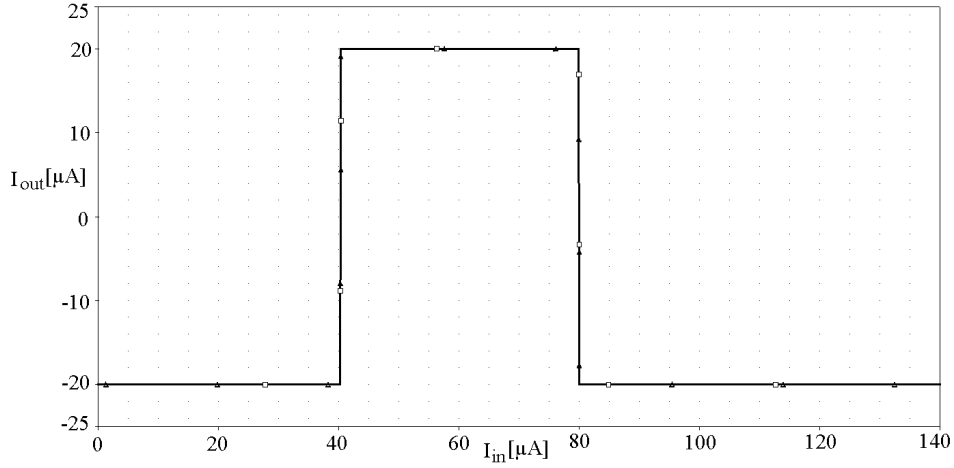
Çıkış katındaki I_o^+ ve I_o^- akımları da V_{out} gerilimi cinsinden aşağıdaki ifadeler ile verilir:

$$I_o^+ = \begin{cases} I_{H1} & V_{out} = V_{DD} \\ -I_{H2} & V_{out} = V_{SS} \end{cases} \quad (2.5)$$

$$I_o^- = \begin{cases} -I_{H2} & V_{out} = V_{DD} \\ I_{H1} & V_{out} = V_{SS} \end{cases} \quad (2.6)$$

Dolayısıyla, tasarlanmış olan devrede giriş katında kullanılan I_1 ve I_2 akımları geçiş karakteristiğinin sıfır olmayan bölgesinin ayarlanmasına olanak tanımaktadır.

Tasarlanmış olan devrenin SPICE benzetimleri için çıkış akımı I_{out} için I_o^+ seçilmiştir ve 0.35 μm AMS CMOS SPICE teknoloji parametreleri kullanılmıştır (parametreler Ek A'da verilmiştir). Besleme gerilimi V_{DD} ve $V_{SS} \pm 1.25$ V olarak alınmış, kutuplama akımı $I_{BIAS}=10 \mu\text{A}$ ve kontrol akımları $I_1=40 \mu\text{A}$, $I_2=80 \mu\text{A}$ ve $I_{H1}=I_{H2}=20 \mu\text{A}$ olarak seçilmesi durumu için I_{out} akımının I_{in} akımı ile değişim karakteristiği Şekil 2.12'de verilmiştir. Çekirdek devre-1 yapısının oluşturulmasında kullanılan tranzistor boyutları Çizelge 2.2'de verilmiştir.

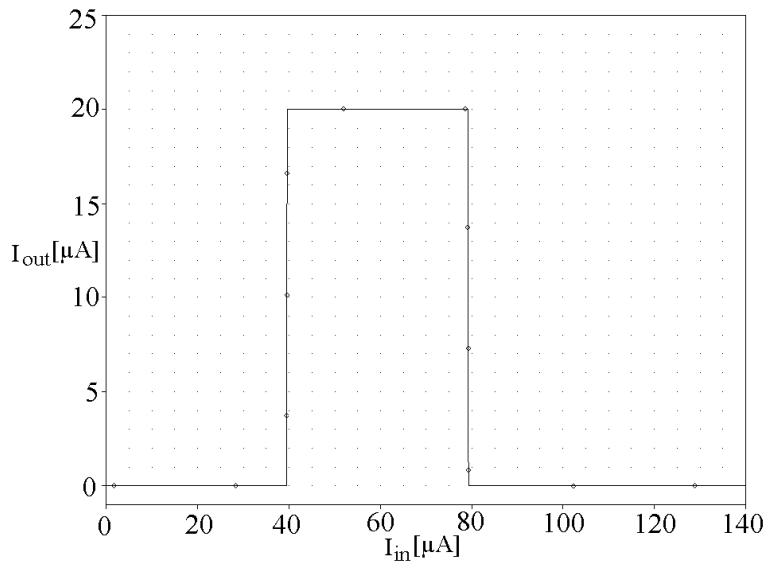


Şekil 2.12 : Çekirdek devre-1 yapısı için I_{out} akımının I_{in} akımı ile değişim karakteristiği ($I_{H2} \neq 0$).

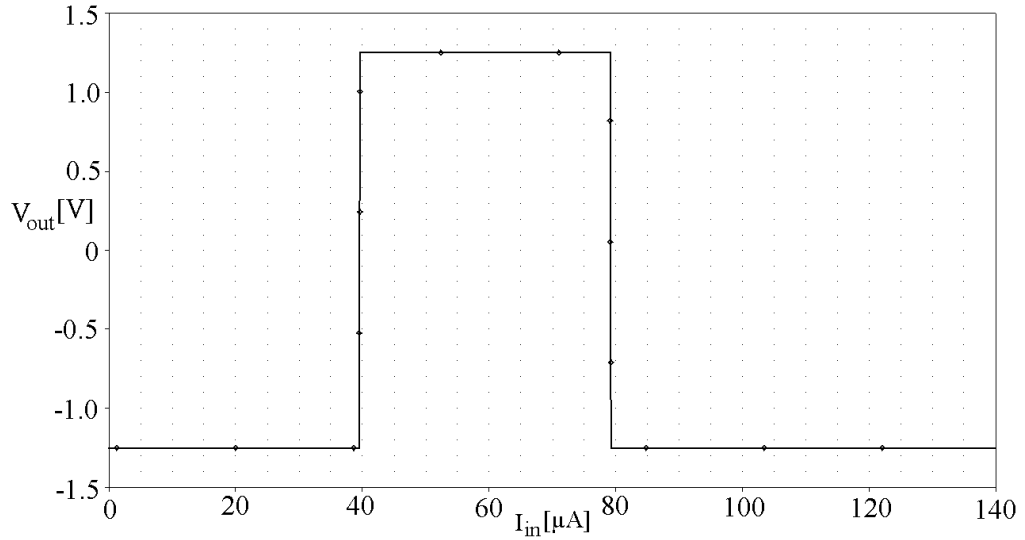
Çalışmanın devamında I_{H2} akımı 0 alınmış, I_{H1} akımı I_H olarak gösterilmiş ve benzetimler yapılmıştır. Kontrol akımları $I_1=40 \mu A$, $I_2=80 \mu A$ ve $I_H=20 \mu A$ alınarak I_{out} akımının I_{in} akımı ile değişim karakteristiği çizdirilmiş ve Şekil 2.13’de verilmiştir. Çıkışın gerilim olması durumu içinse, V_{out} geriliminin I_{in} akımı ile değişim karakteristiği Şekil 2.14’de gösterilmiştir.

Çizelge 2.2 : Çekirdek devre-1 yapısının MOS tranzistor boyutları.

MOSFET	W [μm]	L [μm]
$M_1, M_2, M_3, M_4, M_6, M_7, M_8, M_9, M_{10}, M_{12}$	10.5	0.7
$M_5, M_{11}, M_{13},$	5.25	0.7

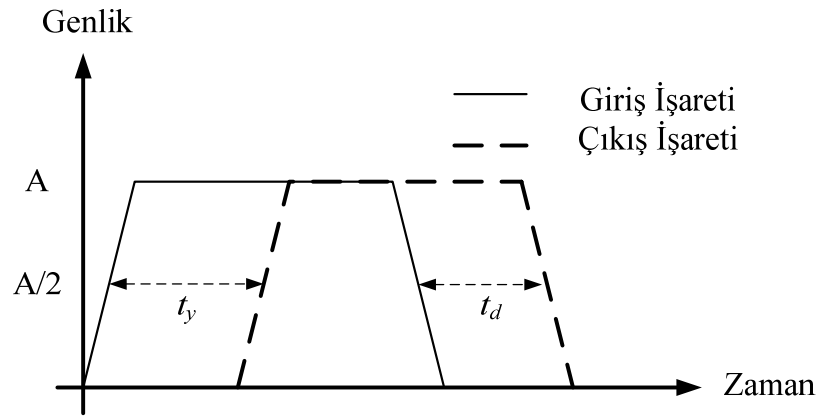


Şekil 2.13 : Çekirdek devre-1 yapısı için I_{out} akımının I_{in} akımı ile değişim karakteristiği ($I_{H2}=0$).



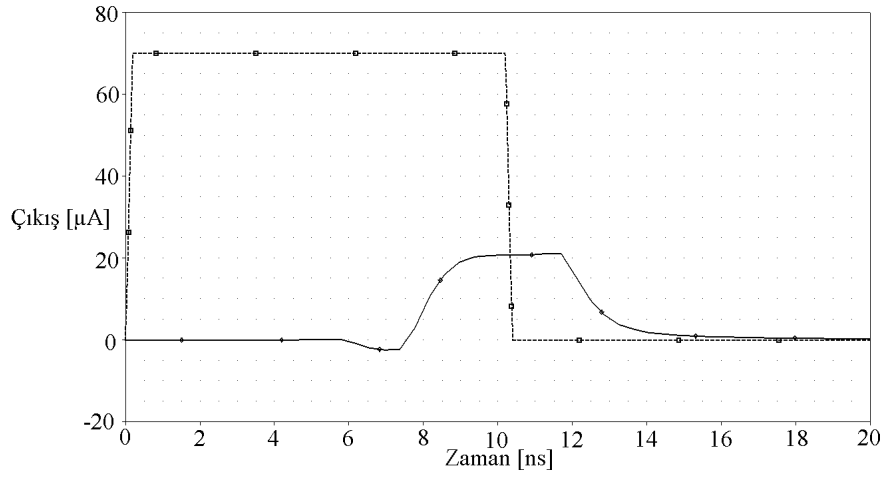
Şekil 2.14 : Çekirdek devre-1 yapısı için V_{out} geriliminin I_{in} akımı ile değişim karakteristiği.

Çekirdek devre-1'in çalışma performansını belirlemek açısından yayılma gecikmesi de incelenmiştir. Yayılma gecikmesi incelemesinde girişe bir kare dalga uygulanmış ve çıkış işareti gözlenmiştir. Devrenin yayılma gecikmesi olarak Şekil 2.15'de gösterilen t_y süresi ile t_d süresinin aritmetik ortalaması kullanılmıştır; t_y süresi için, giriş işareti yükselme durumunda iken genliğinin %50'sine ulaştığı an ile çıkış genliğinin %50'ye ulaştığı an arasındaki fark, t_d ise t_y ile benzer şekilde işaretlerin düşme durumunda hesaplanan fark alınmıştır.



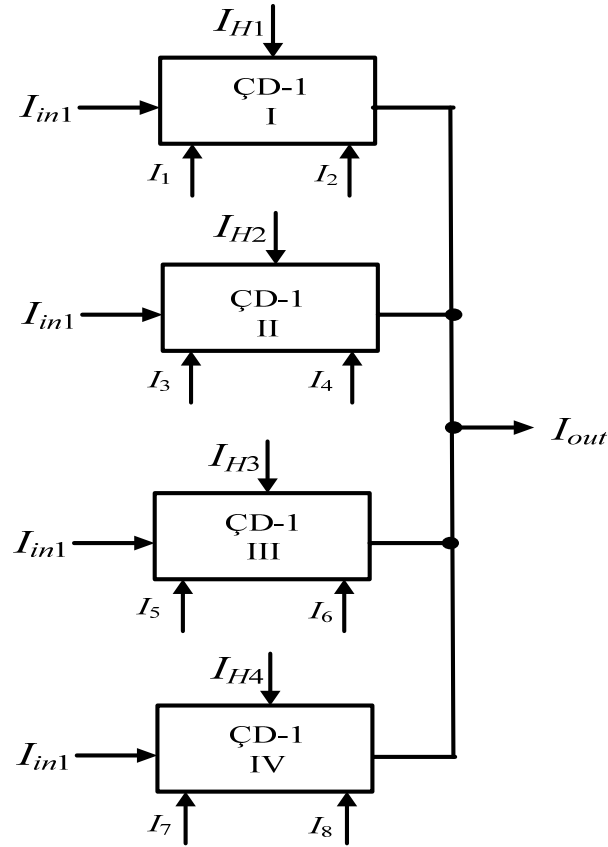
Şekil 2.15 : Giriş-çıkış işareti yayılma gecikmesi.

Çekirdek devre-1'in yayılma gecikmesi benzetimi için kontrol akım parametreleri $I_1=40 \mu A$, $I_2=80 \mu A$ ve $I_H=20 \mu A$ olarak ayarlanmıştır. Girişe genliği $70 \mu A$ olan bir kare dalga uygulanmış ve benzetim sonucunda gecikme, Şekil 2.16'da gösterildiği gibi 5 ns olarak bulunmuştur.



Şekil 2.16 : Çekirdek devre-1'in yayılma gecikmesi benzetim sonucu.

1×1 boyutlu bir sınıflandırıcı devre örneği için, dört adet çekirdek devre-1 yapısı Şekil 2.17'de gösterildiği biçimde paralel olarak bağlanır. Bu tek boyutlu sınıflandırıcı yapısında kontrol akımları ile, uygun bölgeye düşen veri ayırt edilmektedir. Çekirdek devre-1 yapılarına uygulanan kontrol akımları Çizelge 2.3'de verilmiştir.

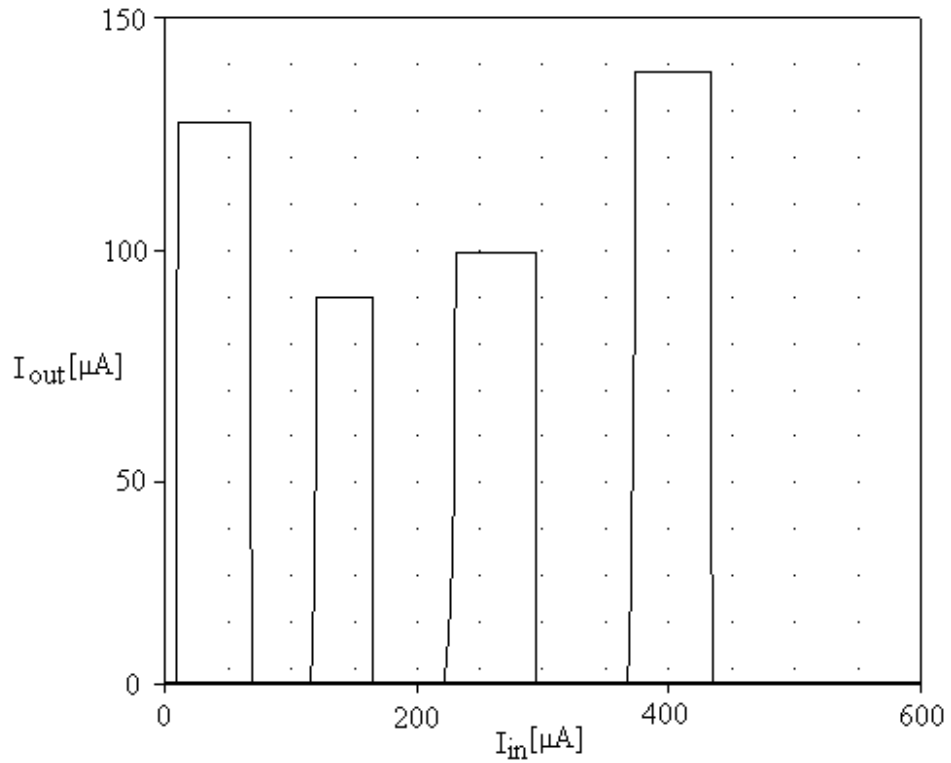


Şekil 2.17 : Çekirdek devre-1 ile gerçekleştirilen 1×1-D sınıflandırıcı blok diyagramı.

Çizelge 2.3 : 1×1-D sınıflandırıcı yapısı çekirdek devre-1 kontrol akımları.

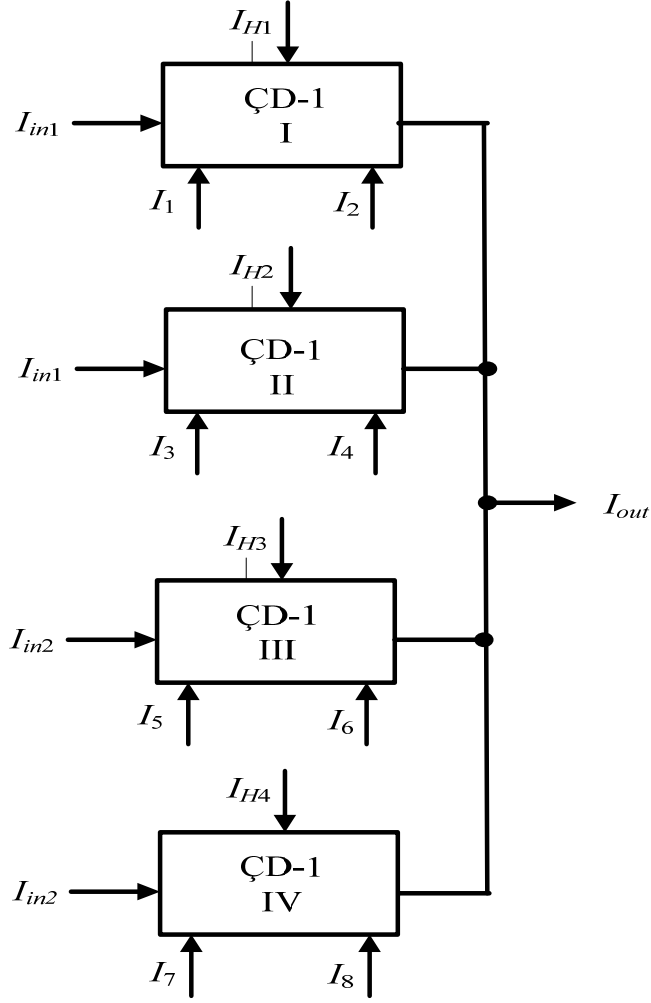
Çekirdek Devre-1	I_1 (μA)	I_2 (μA)	I_H (μA)
I	20	80	130
II	130	180	90
III	230	310	100
IV	370	440	140

Çekirdek devre-1 yapıları ile oluşturulmuş 1×1 boyutlu sınıflandırıcının SPICE benzetimi sonucu elde edilen I_{out} akımının I_{in} akımı ile değişim karakteristiği Şekil 2.18’de verilmiştir.



Şekil 2.18 : ÇD-1 ile gerçekleştirilen 1-D sınıflandırıcı için I_{out} akımının I_{in} akımı ile değişim karakteristiği.

Çekirdek devre-1 yapıları kullanılarak 1×1 boyutlu veriler dışında 2×1 hatta daha büyük $n \times m$ boyutlu veriler de sınıflandırılabilir. Şekil 2.19’da da çok boyutlu verilerin sınıflandırılabilmesi amacıyla benzer topolojiden yararlanarak çekirdek devre-1 blokları ile kurulan yapı verilmiştir. Bu yapıda 4 adet çekirdek devre-1 yapısı birbirine bağlanmış ve çekirdek devre I ve II’ye birinci veri kümesi, III ve IV’e ise ikinci veri kümesi giriş olarak uygulanmıştır. Böylece 2×1 boyutlu veri sınıflandırıcı yapısı oluşturulmuştur.



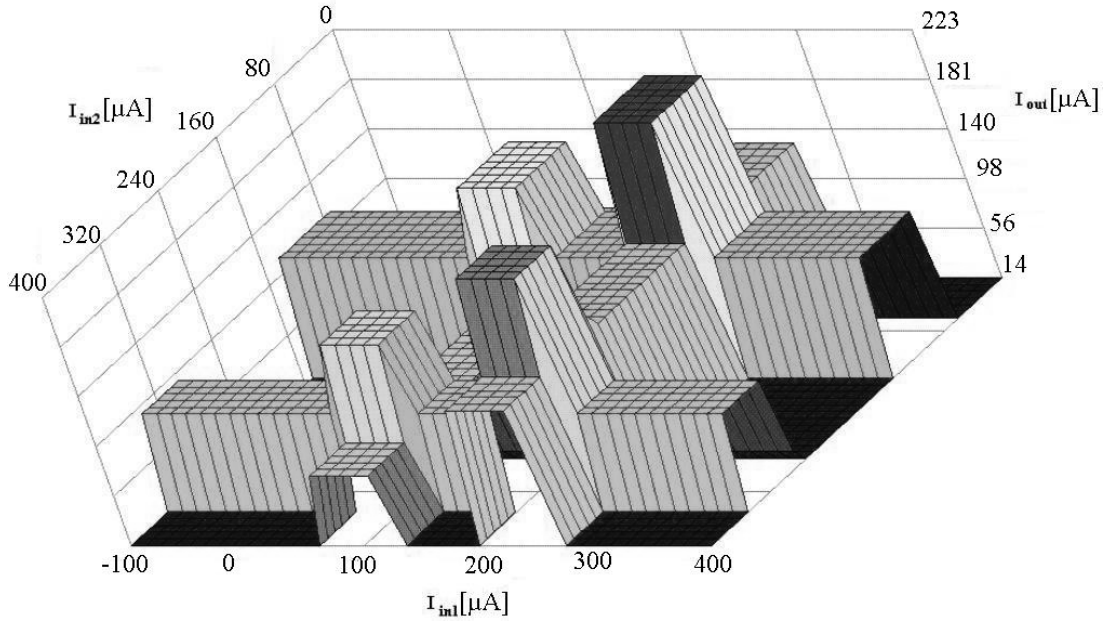
Şekil 2.19 : ÇD-1 ile gerçekleştirilen 2×1-D sınıflandırıcı blok diyagramı.

İki boyutlu sınıflandırıcının benzetimini yapmak amacıyla Çizelge 2.4’de verilen kontrol parametreleri seçilmiş olup her bir çekirdek devre-1’in güç tüketimleri de ayrı ayrı gösterilmiştir. Bu güç tüketimi kontrol akımlarının büyüklüğüne göre değişiklik göstermektedir.

Çizelge 2.4 : 2×1-D sınıflandırıcı yapısı çekirdek devre-1 kontrol akımları.

Çekirdek Devre-1	I_1 (μA)	I_2 (μA)	I_H (μA)	Güç Tüketimi
I	70 μA	140 μA	60 μA	0.46 mW
II	210 μA	280 μA	120 μA	1.03 mW
III	60 μA	140 μA	100 μA	0.51 mW
IV	280 μA	360 μA	80 μA	1.21 mW

Çekirdek devre-1 yapıları ile oluşturulmuş 2×1 boyutlu sınıflandırıcı devresinin $(I_{in1}-I_{in2})-I_{out}$ karakteristiğinin benzetimi Şekil 2.20’de verilmiştir.

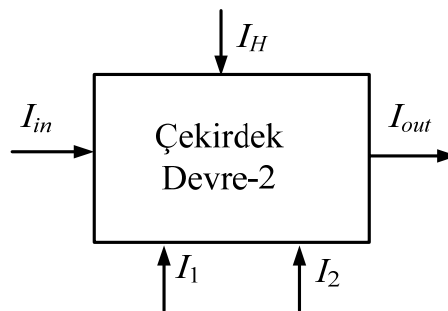


Şekil 2.20 : ÇD-1 ile gerçekleştirilen 2×1 -D sınıflandırıcı ($I_{in1}-I_{in2}$)- I_{out} karakteristiği.

Şekil 2.20’de görüldüğü gibi Şekil 2.19’deki gibi tasarlanmış iki boyutlu sınıflandırıcı devresi ile 8 farklı veri kümesi ayırt edilebilmektedir. Eğer sıfır düzlemi de bir sınıflama sonucu olarak kabul edilirse, 9 farklı sınıfa ait veri sınıflandırılmış olur.

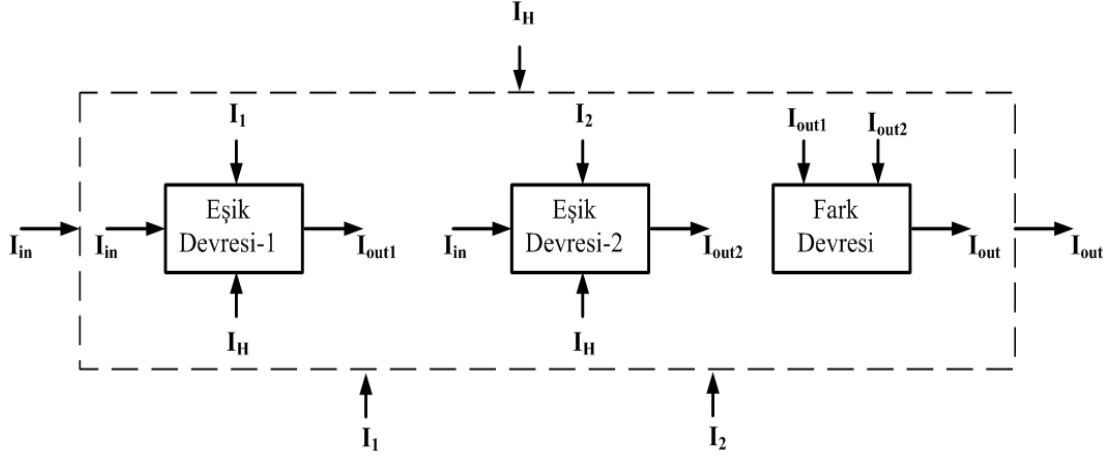
2.2.2 Çekirdek devre-2

Giriş-çıkış karakteristiği Şekil 2.1’de verilen devre farklı bir açıdan yaklaşarak tasarlanmıştır. Devrenin blok yapısı Şekil 2.21’de verilmiştir. Gerçekleştirilen devre ilk yapıdan farklı olarak eşik devreleri kullanılarak oluşturulmuştur.



Şekil 2.21 : Akım-modlu ÇD-2’nin blok yapısı.

Çekirdek devre işlevsel diyagramı Şekil 2.22’de gösterildiği biçimde iki adet eşik devresi ve bir fark devresinden oluşmaktadır.

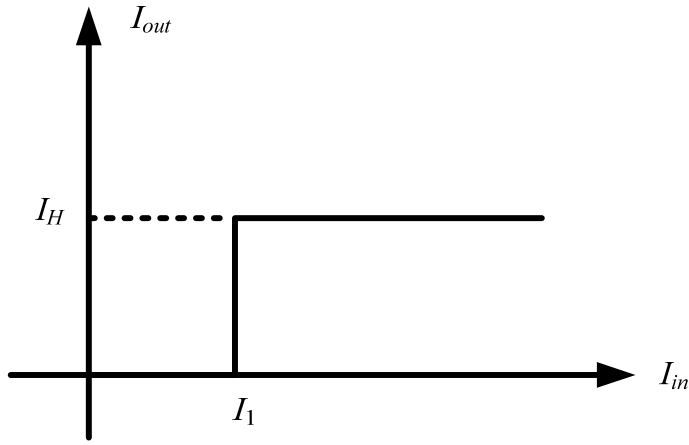


Şekil 2.22 : Akım-modlu ÇD-2 blok diyagramı iç yapısı [55].

Eşik devresinin giriş-çıkış karakteristiği Şekil 2.23’de gösterilmiş ve tanım bağıntısı

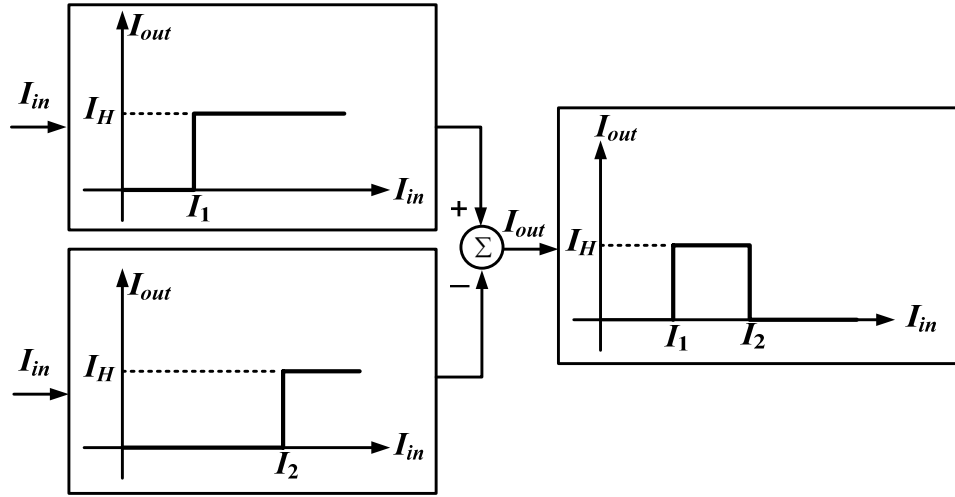
$$I_{out} = \begin{cases} I_H & I_1 < I_{in} \\ 0 & \text{diğer hallerde} \end{cases} \quad (2.7)$$

şeklinde verilmiştir.



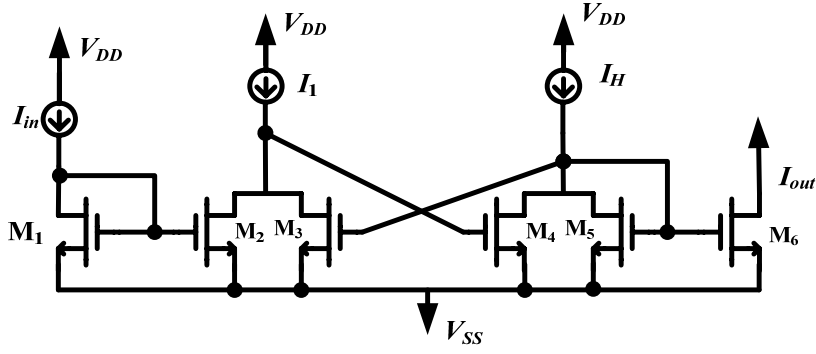
Şekil 2.23 : Eşik devresinin giriş-çıkış karakteristiği.

Çekirdek devre-2 yapısının gerçekleştirilmesi için Şekil 2.24’de verilen blok diyagram kullanılmıştır. Bu yapı da iki eşik devresi ve bir fark alma devresinden oluşmaktadır. İki eşik devresinin çıkış akımlarının farkı alınarak Şekil 2.29’da gösterildiği gibi sınıflandırma bölgesi oluşturulmuştur. Bu bölgenin oluşturulması için eşik devrelerinin I_H akımları eşit seçilmeli ve $I_2 > I_1$ sağlanmalıdır.

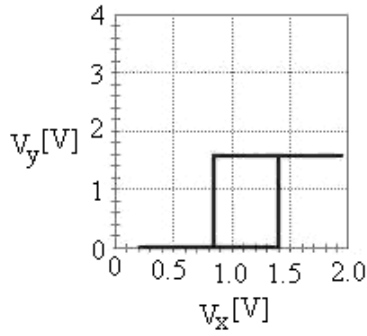


Şekil 2.24 : ÇD-2 yapısının gerçekleştirilme blok şeması.

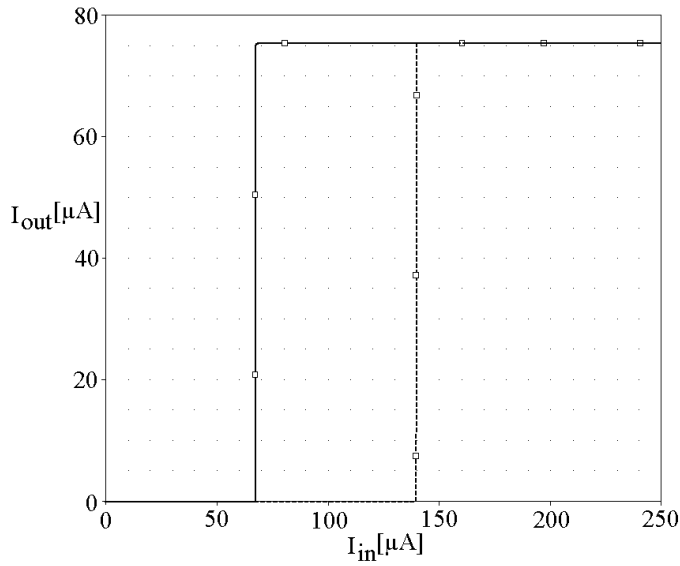
Çekirdek devre-2'nin gerçekleştirilmesi için kullanılan eşik devresi Şekil 2.25 ile gösterilmiştir [56]. Devrede I_1 akımı eşik değeridir. Eğer giriş akımı I_{in} , I_1 akımından büyükse devrenin çıkışındaki akım I_H ve M_3 tranzistorunun savak gerilimi yaklaşık olarak V_{SS} olur; giriş akımı I_{in} eşik akımından küçük olduğu zaman ise M_3 tranzistorunun savak gerilimi yaklaşık olarak V_{DD} 'ye eşittir. Diğer taraftan M_4 tranzistorunun akımı M_3 tranzistorunun savak gerilimi ile de kontrol edilmektedir. M_4 ve M_5 (M_4 açıkken M_5 kapalı, M_5 açık iken M_4 kapalı) tranzistorları anahtar gibi çalıştıklarından I_{in} akımının değerine göre I_H akımı M_4 ya da M_5 tranzistorundan akmaktadır. Şekil 2.25'deki devre CD4007 CMOS tranzistor entegresi kullanılarak sınanmış ve osiloskop ekranında kullanılan eşik devresinde histerezis karakteristiği olduğu gözlenmiştir. Devrenin test düzeneği kurulurken akım kaynakları yerine direnç bağlanmıştır. Bu dirençlerin uçlarına uygulanan gerilim ile kontrol akım ve giriş akım değerleri elde edilmiştir. Pozitif geribeslemeli eşik devresini sınamak için I_1 akımı $R_1=20\text{ k}\Omega$ direnç üzerinden $V_{R1}=1.7\text{ V}$ olacak şekilde uygulanmıştır. Benzer şekilde $R_H=20\text{ k}\Omega$, $V_H=1.5\text{ V}$ alınarak uygulanmış ve böylece uygun akım değerleri $I_1=85\text{ }\mu\text{A}$ ve $I_H=75\text{ }\mu\text{A}$ olacak şekilde sağlanmıştır. Çıkışa bağlanan $100\text{ k}\Omega$ 'luk direnç ile akımın değişimi osiloskopta gözlenerek akım değeri ölçülmüştür. Kurulan devrenin giriş-çıkış karakteristiği test sonuçları osiloskobun X-Y özelliği kullanılarak Şekil 2.26'da verilmiştir. Bu şekilde X eksenini için osiloskop skalası 0.5 V/div , Y eksenini içinse 1 V/div seçilmiştir. Devrenin benzetim sonucu için kontrol akımları $I_1=85\text{ }\mu\text{A}$ ve $I_H=75\text{ }\mu\text{A}$ olacak şekilde ayarlanmış, devrenin aynı histerezis davranışını gösterdiği, giriş önce arttırılarak, sonra da azaltılarak uygulandığında elde edilmiş ve sonucu Şekil 2.27'de verilmiştir.



Şekil 2.25 : Pozitif geribeslemeli eşik devresi devre şeması.

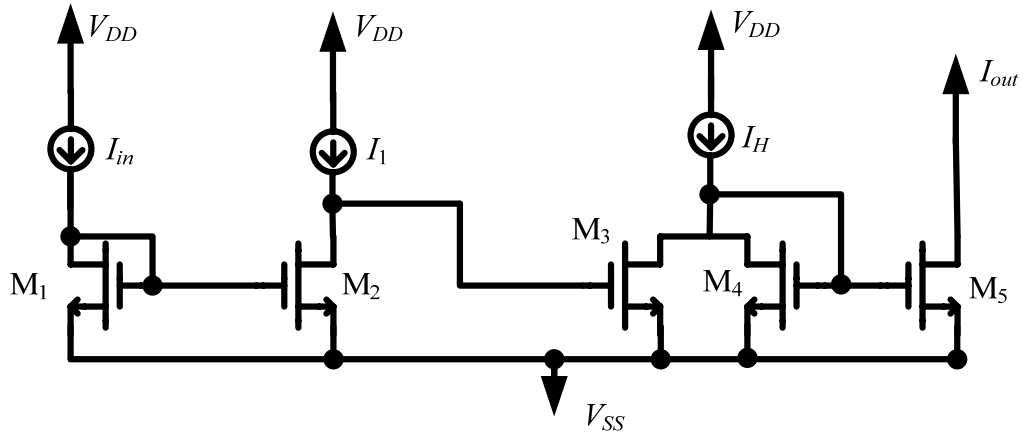


Şekil 2.26 : Pozitif geribeslemeli eşik devresi osiloskop çıktısı.



Şekil 2.27 : Pozitif geribeslemeli eşik devresi benzetim karakteristiği.

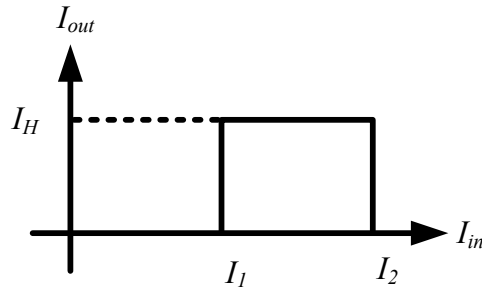
Şekil 2.25'deki devrede pozitif geribeslemeyi sağlayan M_3 tranzistörü çıkartılarak histerezis karakteristiğinin oluşmamasının sağlandığı hem donanım hem benzetimle sınanmış, gerçekleştirilen yeni eşik devresinin şeması Şekil 2.28'de sunulmuştur. Bu yapıda kullanılan kaynaklar basit akım aynaları olarak devreye uygulanmıştır.



Şekil 2.28 : Eşik devresi devre şeması.

Devrede I_1 akımı eşik değeridir. Eğer giriş akımı I_{in} , I_1 akımından büyükse devrenin çıkışındaki akım I_H ve M_2 tranzistorunun savak gerilimi yaklaşık olarak V_{SS} olur. Giriş akımı I_{in} eşik akımından küçük olduğu zaman da M_2 tranzistorunun savak gerilimi yaklaşık olarak V_{DD} 'ye eşittir. Diğer taraftan M_3 tranzistorunun akımı M_2 tranzistorunun savak gerilimi ile de kontrol edilmektedir. M_2 tranzistorunun savak gerilimi yaklaşık V_{SS} olduğu zaman M_3 tranzistoru kesime girerek I_H akımının M_4 üzerinden akması sağlanır. Benzer şekilde M_2 tranzistorunun savak gerilimi yaklaşık V_{DD} olduğu zaman M_3 tranzistoru iletme geçerek I_H akımının M_3 üzerinden akması sağlanır. M_3 ve M_4 (M_3 açıkken M_4 kapalı, M_4 açık iken M_3 kapalı) tranzistorları anahtar gibi çalıştıklarından I_{in} akımının değerine göre I_H akımı M_3 ya da M_4 tranzistorundan akmaktadır.

Böylece eşik devreleri kullanılarak gerçekleştirilmiş olan çekirdek devre-2 yapısının giriş-çıkış karakteristiği Şekil 2.29'da verilmiştir.



Şekil 2.29 : ÇD-2 sınıflandırıcı devresi giriş-çıkış karakteristiği.

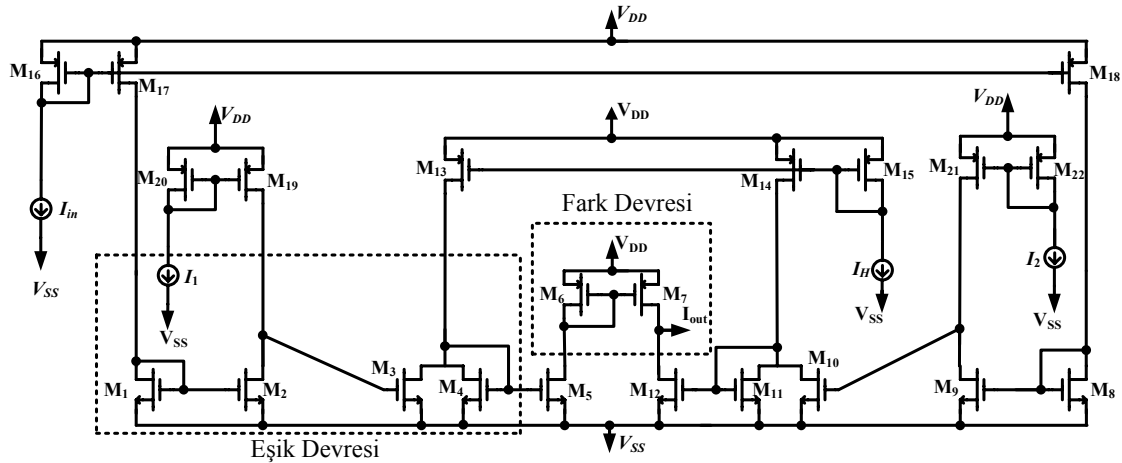
Giriş-çıkış karakteristiği Şekil 2.29'da verilen devrenin tanım bağıntısı da,

$$I_{out} = \begin{cases} I_H & I_1 < I_{in} < I_2 \\ 0 & \text{diğer hallerde} \end{cases} \quad (2.8)$$

şeklinde verilir.

Çekirdek devre-2 yapısının şeması Şekil 2.30'da verilmiştir. Çekirdek devre-2 yapısında kullanılan kontrol ve giriş akımları basit akım aynası kullanılarak devreye uygulanmıştır. Şekil 2.30'da M_1 - M_4 ve M_8 - M_{11} tranzistorları eşik devrelerini oluşturmaktadır. Çıkış akımlarının farkını almak için kullanılan fark devresi ise basit akım aynası (M_6 ve M_7) kullanılarak gerçekleştirilmiştir. M_{13} , M_{14} ve M_{15} tranzistorları eşik devrelerine aynı I_H akımını uygulamak için kullanılmıştır. Benzer şekilde M_{16} , M_{17} ve M_{18} tranzistorları da girişlere aynı I_{in} akımını uygulamak için kullanılmıştır.

Önerilen çekirdek devre-2 yapısının benzetiminde 0.35 μm AMS CMOS teknoloji parametreleri kullanılmış ve besleme gerilimleri V_{DD} ve $V_{SS} \pm 1.65$ V olarak alınmıştır. Tranzistor boyutları Çizelge 2.5'de verilmiştir.

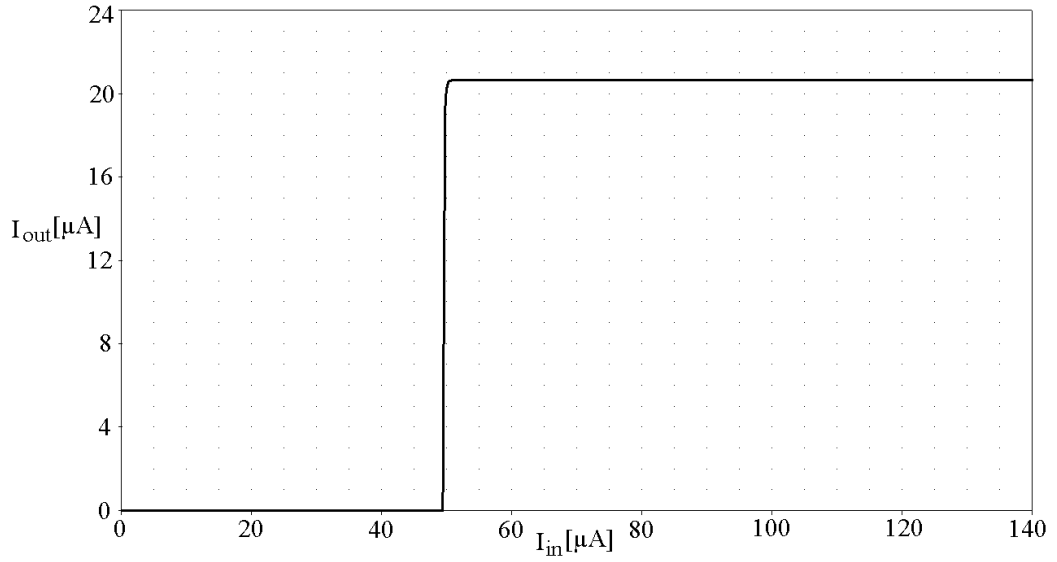


Şekil 2.30 : ÇD-2 devre şeması [55].

Çizelge 2.5 : ÇD-2'de kullanılan MOS tranzistor boyutları.

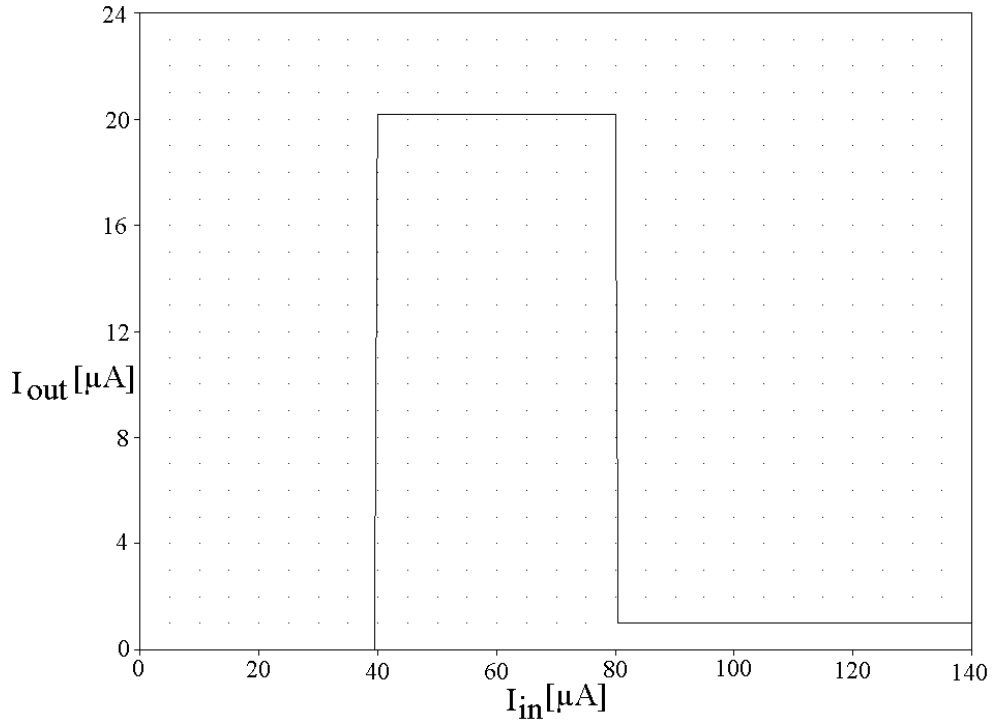
MOSFET	W [μm]	L [μm]
$M_1, M_2, M_3, M_4, M_5, M_8, M_9, M_{10}, M_{11}, M_{12}$	10.5	1.05
$M_6, M_7, M_{13}, M_{14}, M_{15}, M_{16}, M_{17}, M_{18}, M_{19}, M_{20}, M_{21}, M_{22}$	35.5	1.05

$I_1=50$ μA ve $I_H=20$ μA için eşik devresinin benzetim sonuçları Şekil 2.31'de verilmiştir.



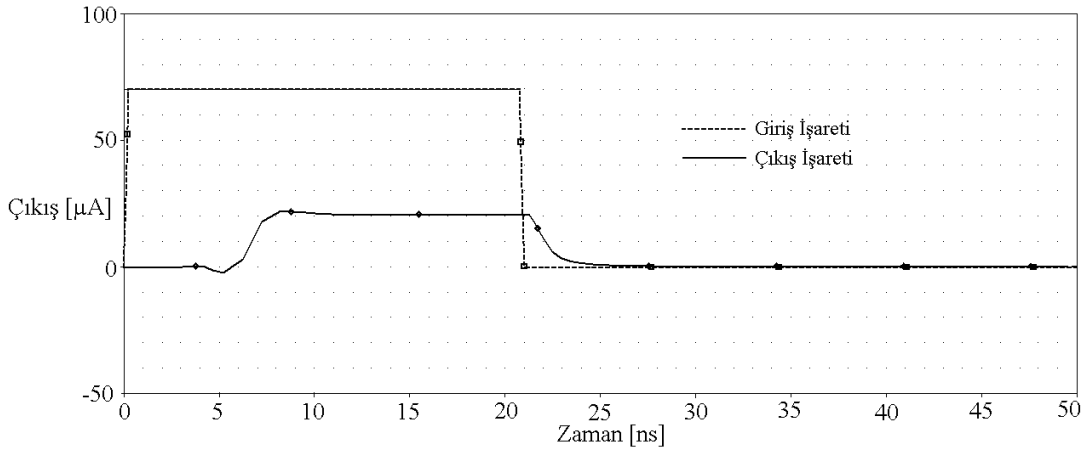
Şekil 2.31 : Eşik devresinin giriş-çıkış karakteristiği.

Çekirdek devre-2 yapısının benzetim sonuçları $I_1=40 \mu A$, $I_2=80 \mu A$ ve $I_H=20 \mu A$ için Şekil 2.32’de gösterilmiştir. Devrenin güç tüketimi 0.38 mW olarak elde edilmiştir. Bu güç tüketimi kontrol akımlarının büyüklüğüne göre değişiklik göstermektedir.



Şekil 2.32 : ÇD-2 yapısı giriş-çıkış karakteristiği.

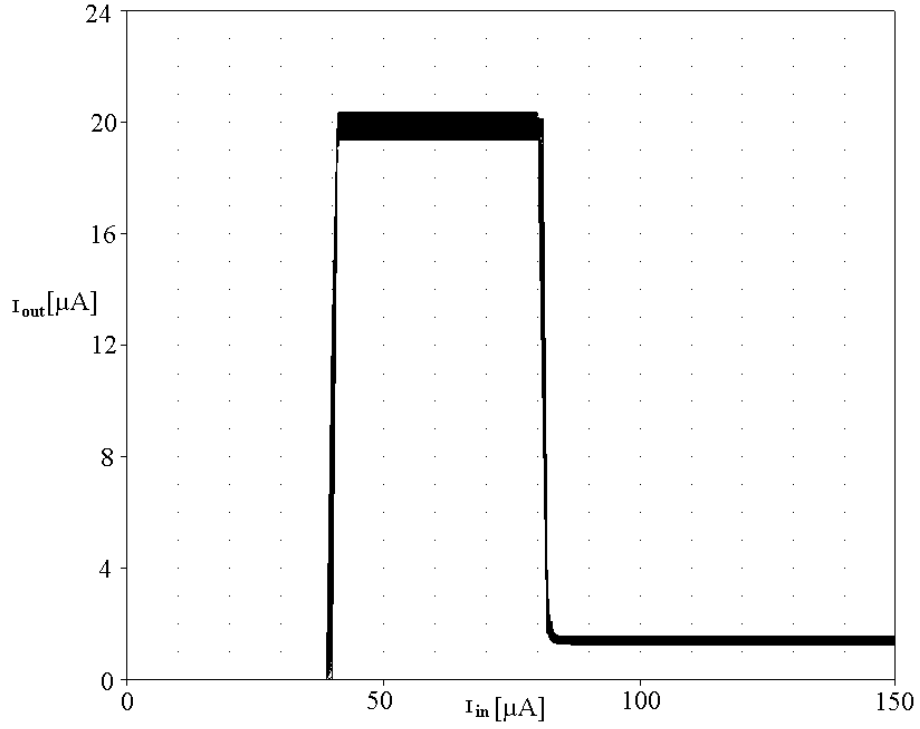
Çekirdek devre-2’nin yayılma gecikmesini veren benzetim için kontrol akım parametreleri $I_1=40 \mu A$, $I_2=80 \mu A$ ve $I_H=20 \mu A$ olarak ayarlanmış ve girişe genliği 70 μA olan bir kare dalga uygulanmıştır. Benzetim sonucunda, gecikme Şekil 2.33’den hesaplanmış ve 4 ns olarak bulunmuştur.



Şekil 2.33 : ÇD-2 yapısı yayılma gecikmesi benzetimi.

Çekirdek devre-2 yapısında kontrol akım parametrelerinin $I_1=10 \mu\text{A}$, $I_2=20 \mu\text{A}$ ve $I_H=2 \mu\text{A}$ olarak ayarlanması durumunda, yayılma gecikmesi 28 ns olarak bulunmuştur. Bu yayılma gecikmesinin, kontrol akım değerlerinin küçük seçilmesi ile artmasındaki sebep, ÇD-2 yapısında bulunan parazitik kapasitelerin, yüksek kontrol akım değerleriyle daha hızlı dolması ve düşük kontrol akım değerleriyle daha yavaş dolması şeklinde açıklanabilir.

Şekil 2.30'da verilen devrede özellikle M_{13} ve M_{14} tranzistorlarının eşleşme problemi sonucu, fark devresi çıkışında çıkış akımının I_H değerinde beklenenden hatalı sonuçlar elde edilebilir. Bu hata aynı zamanda üretim sırasında devredeki tranzistorların kanal boyu ve kanal genişliğinde oluşabilecek sapmalardan da kaynaklanmaktadır. I_H çıkış akımı değerindeki sapmaları görebilmek için devredeki tüm tranzistorların kanal boyu uzunluğu L parametresinde $0.11 \mu\text{m}$ ve kanal genişliği W parametresinde $0.4 \mu\text{m}$ 'lik sapma durumunda 100 farklı değer göz önüne alınarak Monte Carlo analizi yapılmıştır. Benzetim için kontrol akımları I_H , I_1 ve I_2 $0 \mu\text{A}$ ile $200 \mu\text{A}$ arasında bir çok nominal akım için test edilmiştir. Bu test sonucunda I_H akımı için nominal değerinden maksimum sapma %3.1, I_1 ve I_2 akımları için de maksimum sapma %2.3 olarak gözlenmiştir. Diğer taraftan akım değeri $0 \mu\text{A}$ olması gereken bölgede ($I_{in} > I_2$) ise akımın değeri minimum $1.1 \mu\text{A}$, maksimum $1.5 \mu\text{A}$ olmaktadır. Ayrıca örnek olarak $I_H=20 \mu\text{A}$, $I_1=40 \mu\text{A}$ ve $I_2=80 \mu\text{A}$ değerleri için Monte Carlo analizi giriş-çıkış karakteristiği benzetim sonucu Şekil 2.34'de verilmiştir. Sınıflandırma uygulamalarında bu sapma değerleri göz önüne alınarak kontrol akımların seçilmesi durumunda yanlış sınıflandırmaya neden olabilecek sapmalar engellenmiş olur.



Şekil 2.34 : ÇD-2 yapısı giriş-çıkış karakteristiği Monte Carlo analizi.

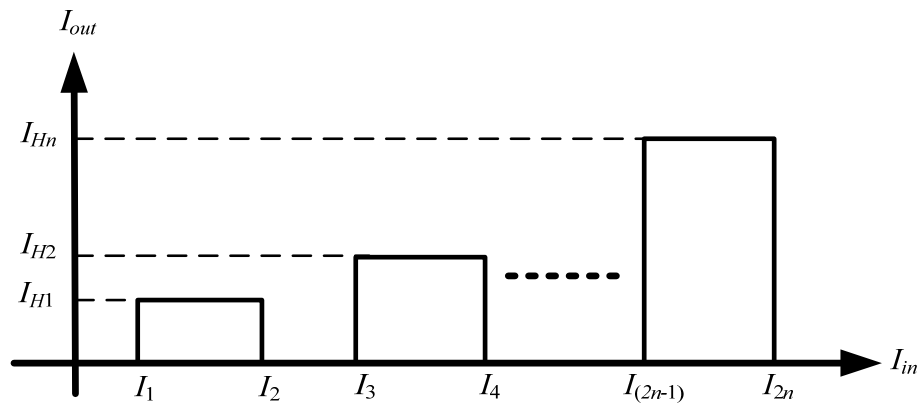
Tek boyutlu verilerin sınıflandırılması için, Şekil 2.17’de gösterildiği gibi, çekirdek devre-2 yapıları paralel olarak bağlanır ve

$$I_{in1} = I_{in2} = \dots = I_{inn} = I_{in} \quad (2.9)$$

$$I_1 < I_2 < I_3 < \dots < I_{(2n-1)} < I_{2n} \quad (2.10)$$

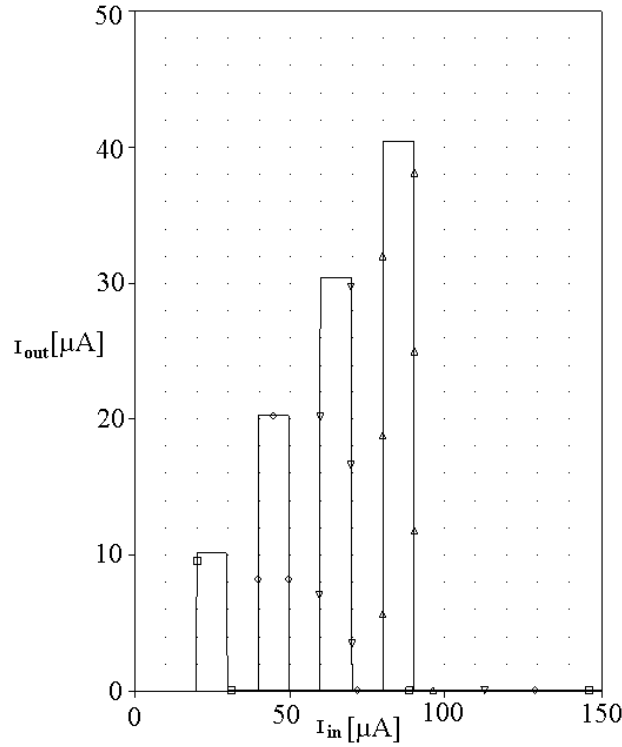
ifadeleri ile verilen koşullar sağlanacak şekilde kontrol akımları seçilir.

Kontrol akımlarının uygun olarak seçildiği durumda 1×1 boyutlu sınıflandırıcının giriş-çıkış karakteristiği Şekil 2.35’de verildiği biçimde elde edilir.



Şekil 2.35 : Tek boyutlu sınıflandırıcının giriş-çıkış karakteristiği.

1×1 boyutlu devrenin benzetimi için 4 adet çekirdek devre-2 yapısı paralel olarak bağlanmıştır. Benzetim sonucu Şekil 2.36’da ve kontrol akımları da Çizelge 2.6’da verilmiştir. Benzetim sonuçlarından görüldüğü gibi “0” bölgesi de dahil beş farklı tipte veri sınıflandırılmaktadır. Devrenin toplam güç tüketimi 0.95 mW’tır.



Şekil 2.36 : ÇD-2 ile gerçekleştirilen 1×1-D sınıflandırıcı giriş-çıkış benzetimi.

Çizelge 2.6 : ÇD-2 ile gerçekleştirilen 1×1-D kontrol akımları (μA olarak).

ÇD-1			ÇD-2			ÇD-3			ÇD-4		
I_1	I_2	I_{H1}	I_3	I_4	I_{H2}	I_5	I_6	I_{H3}	I_7	I_8	I_{H4}
20	30	10	40	50	20	60	70	30	80	90	40

İki boyutlu sınıflandırıcı devrenin benzetimi için Şekil 2.19’da verilen blok şeması çekirdek devre-2 yapıları kullanılarak kurulmuştur. Benzetimde kullanılan her bir çekirdek devre-2 yapısının kontrol akım ve güç tüketim değerleri Çizelge 2.7’de verilmiştir.

Çizelge 2.7 : ÇD-2 ile gerçekleştirilen 2×1-D sınıflandırıcı devresi kontrol akımları (akımlar μA olarak ifade edilmiştir).

Çekirdek Devre-1	$I_1 (\mu A)$	$I_2 (\mu A)$	$I_H (\mu A)$	Güç Tüketimi
I	70 μA	140 μA	60 μA	0.86 mW
II	210 μA	280 μA	120 μA	1.83 mW
III	60 μA	140 μA	100 μA	1.19 mW
IV	280 μA	360 μA	80 μA	1.75 mW

Çekirdek devre-2 ile gerçekleştirilen 2×1-D sınıflandırıcı yapısının SPICE benzetim sonucu Şekil 2.20’dekine benzer biçimde elde edilir. Çekirdek yapılar paralel bağlanarak daha fazla sayıda verinin sınıflandırılması sağlanabilir.

2.2.2.1 Çekirdek devre-2’nin zayıf evirtim benzetimleri

Son zamanlarda zayıf evirtimde çalışan analog devreler özellikle güç tüketiminin küçük olmasının önemli olduğu uygulamalarda önem kazanmıştır. Bu uygulamaların başında ise pille beslenen, az güç tüketen ve küçük boyutlu tıbbi cihazlar gelmektedir. MOS tranzistorların zayıf evirtimde çalıştırılması sayesinde kurulan devrelerin güç tüketimleri μW ’lar mertebesinde olmaktadır.

MOS tranzistorda, V_{GS} geçit-kaynak geriliminin V_T eşik geriliminden büyük olması halinde ($V_{DS} \geq V_{GS} - V_T$, $V_{GS} > V_T$), tranzistorun akım gerilim ilişkisi doyma bölgesinde

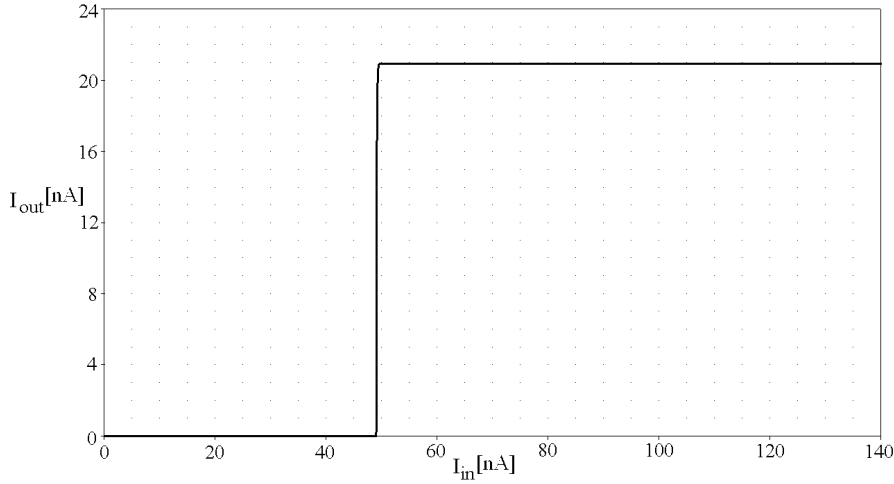
$$I_D = \frac{k}{2} (V_{GS} - V_T)^2 \quad (2.11)$$

karesel bağıntısı ile verilir. Bu bağıntıdaki k MOS tranzistorun geçiş iletkenlik parametresidir. Genellikle, $V_{GS} < V_T$ için I_D savak akımı ihmal edilir. Gerçekte, $V_{GS} < V_T$ için yüzeye yakın bölgelerde bir elektron yoğunluğu bulunduğundan, savak akımı sıfır değildir. Bu bölgeye zayıf evirtim bölgesi denir. Başka bir deyişle, geçit kaynak geriliminin eşik geriliminin üzerinde olduğu bölge kuvvetli evirtim bölgesi, bu gerilimin eşik geriliminin altında olduğu bölge de zayıf evirtim bölgesidir. V_{GS} gerilimi V_T eşik gerilimine yaklaştıkça, MOS tranzistorun I_D - V_{GS} karakteristiği karesel bağımlılıktan üstel bağımlılığa dönüşür. Bir MOS tranzistorun zayıf evirtimde akıtacağı savak akımı

$$I_D = I_s e^{\left(\frac{q(V_{GS} - V_T - V_{offset})}{nkT} \right)} \left(1 - e^{-\frac{qV_{DS}}{kT}} \right) \quad (2.12)$$

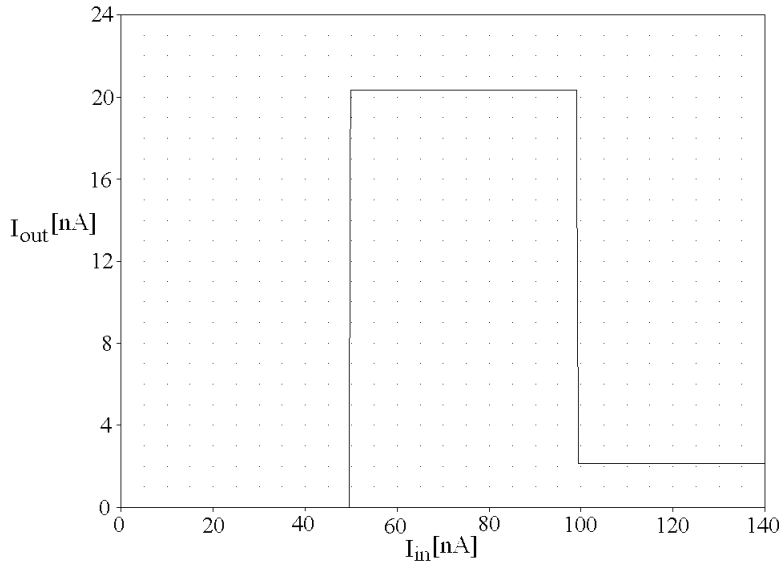
bağıntısıyla verilebilir. Bu bağıntıda V_{GS} , V_{DS} büyüklükleri sırası ile geçit-kaynak ve savak-kaynak gerilimlerini vermektedir. I_s akım katsayısı, T sıcaklık, V_{offset} gerilimi - 0.1 ile 0.1 V arasında değişen bazı gerilim terimlerinin toplamı, n zayıf evirtim salınım parametresi, k Boltzmann sabiti ve q bir elektronun yüküdür [57].

Çekirdek devre-2'nin oluşturulmasında kullanılan temel blok olan eşik devresinin zayıf evirtimde çalıştırılması için, kontrol akımları $I_1=50$ nA ve $I_H=20$ nA olarak seçilmiş ve eşik devresinin benzetim sonucu Şekil 2.37'de verilmiştir [58].



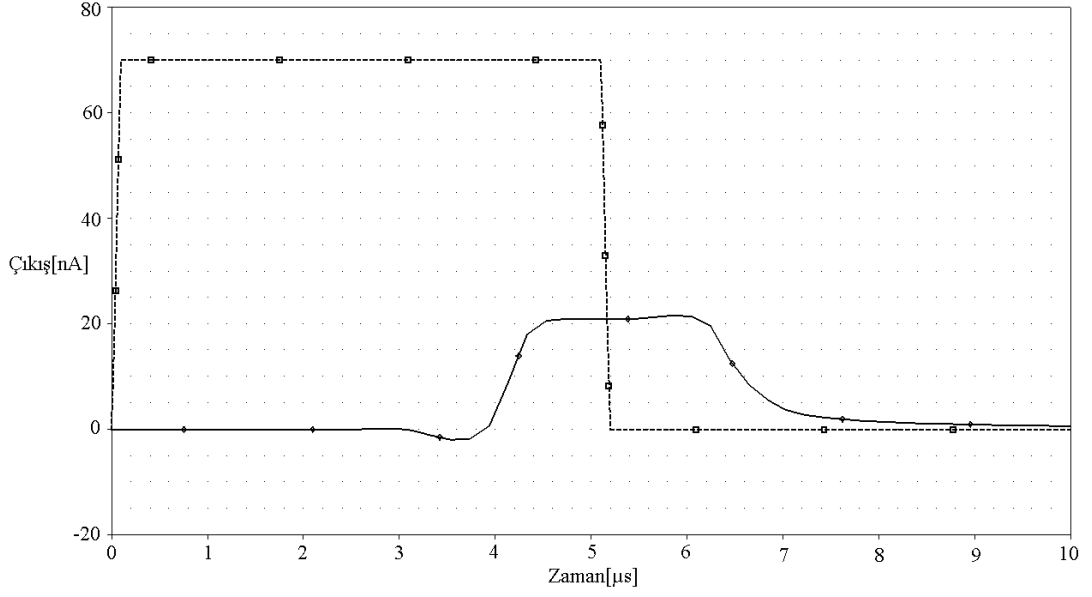
Şekil 2.37 : Zayıf evirtimde çalışan eşik devresi giriş-çıkış karakteristiği.

Çekirdek devre-2 yapısının benzetim sonuçları $I_1=50$ nA, $I_2=100$ nA ve $I_H=20$ nA için Şekil 2.38'de verilmiş ve güç tüketimi $0.43 \mu W$ olarak elde edilmiştir. Bu güç tüketimi kontrol akımlarının büyüklüğüne göre değişiklik göstermektedir.



Şekil 2.38 : Zayıf evirtimde çalışan ÇD-2 yapısı giriş-çıkış karakteristiği.

Benzer şekilde çekirdek devre-2'nin yayılma gecikmesi benzetimi için kontrol akım parametreleri $I_1=50$ nA, $I_2=100$ nA ve $I_H=20$ nA olarak ayarlanmış ve girişe genliği 70 nA olan bir kare dalga uygulanmıştır. Benzetim sonucunda, Şekil 2.39'dan hesaplanan giriş-çıkış dalga şekilleri arasındaki yayılma gecikmesi $2.8 \mu s$ olarak bulunmuştur.

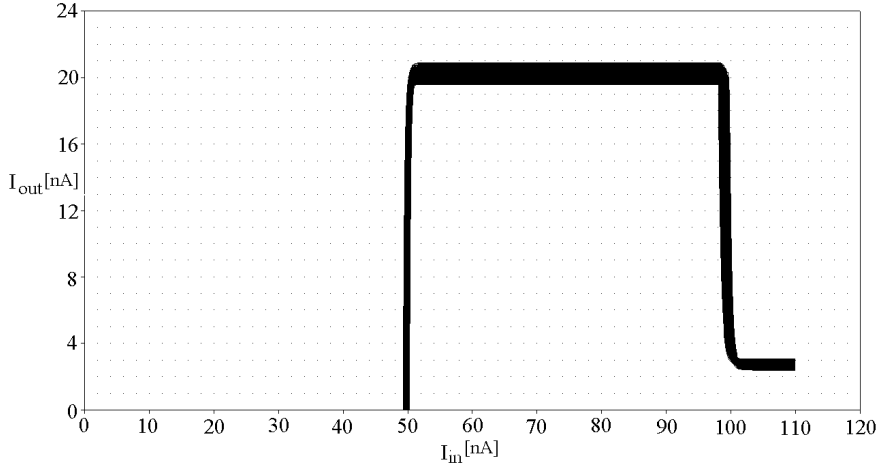


Şekil 2.39 : Zayıf evirtimde çalışan ÇD-2 için yayılma gecikmesi benzetimi.

Şekil 2.30’da verilen çekirdek devre-2 yapısındaki tranzistorlar zayıf evirtimde çalıştırıldığında M_{13} ve M_{14} tranzistorlarının eşleşme problemi sonucu fark devresi çıkışında, çıkış akımının I_H değerinde beklenenden hatalı sonuçlar elde edilebilir. Bu hata aynı zamanda üretim sırasında devredeki diğer tranzistorların kanal boyu ve kanal genişliğinde oluşabilecek sapmalardan da kaynaklanır. I_H çıkış akımı değerindeki sapmaları görebilmek için devredeki tüm tranzistorların kanal boyu uzunluğu L parametresinde $0.11 \mu\text{m}$ ve kanal genişliği W parametresinde $0.4 \mu\text{m}$ ’lik sapma durumunda 100 farklı değer göz önüne alınarak Monte Carlo analizi yapılmıştır.

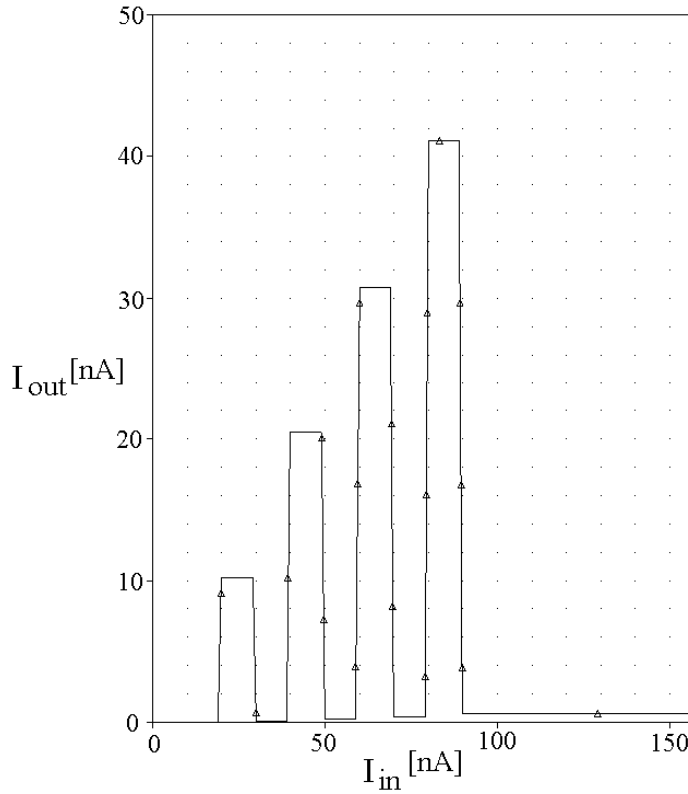
Benzetim için kontrol akımları I_H , I_1 ve I_2 0 nA ile 200 nA arasında bir çok nominal akım için test edilmiştir. Bu test sonucunda I_H akımı için nominal değerinden maksimum sapma %5.5, I_1 ve I_2 akımları için de maksimum sapma %4.2 olarak gözlenmiştir. Diğer taraftan akım değeri 0 nA olması gereken bölgede ($I_{in} > I_2$) ise akımın değeri minimum 2.3 nA, maksimum 3.3 nA olmaktadır.

Ayrıca örnek olarak $I_H=20 \text{ nA}$, $I_1=50 \text{ nA}$ ve $I_2=100 \text{ nA}$ değerleri için Monte Carlo analizi giriş-çıkış karakteristiği benzetim sonucu Şekil 2.40’da verilmiştir. Dolayısıyla sınıflandırma uygulamalarında bu sapma değerleri göz önüne alınarak kontrol akımların seçilmesi durumunda yanlış sınıflandırmaya neden olabilecek sapmalar engellenmiş olur.



Şekil 2.40 : Zayıf evirtimde çalışan ÇD-2 yapısı giriş-çıkış karakteristiği Monte Carlo analizi.

1×1 boyutlu sınıflandırıcı devrenin benzetimi için dört adet çekirdek devre-2 yapısı paralel olarak Şekil 2.17'deki biçimde bağlanmıştır. Benzetim sonucu Şekil 2.41'de ve çekirdek devre-2 yapılarının kontrol akım ve güç tüketim değerleri de Çizelge 2.8'de verilmiştir. Şekil 2.41'deki benzetim sonuçlarından görüldüğü gibi x -ekseni de dahil olmak üzere beş farklı tipte veri sınıflandırılmaktadır; devrenin toplam güç tüketimi 1.62 μ W çıkmaktadır.



Şekil 2.41 : Zayıf evirtimde çalışan ÇD-2 yapısı ile gerçekleştirilen 1-D sınıflandırıcı giriş-çıkış karakteristiği.

2×1 boyutlu sınıflandırıcı devrenin benzetimi için Şekil 2.19'daki yapı, çekirdek devre-2 blokları kullanılarak kurulmuştur. Benzetimde kullanılan her çekirdek devre-2'nin kontrol akımları ve güç tüketimleri Çizelge 2.9'da verilmiştir.

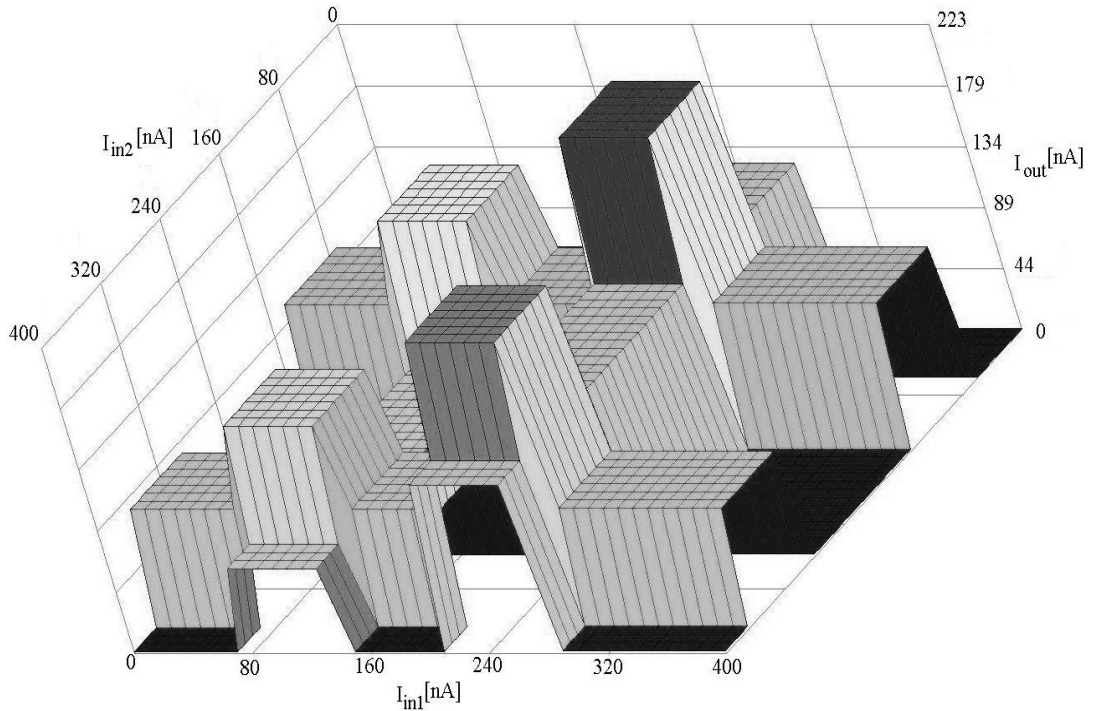
Çizelge 2.8 : Zayıf evirtimde çalışan ÇD-2 yapısı ile gerçekleştirilen 1-D sınıflandırıcı yapısı kontrol akımları (akımlar nA).

ÇD-1			ÇD-2			ÇD-3			ÇD-4		
I_1	I_2	I_{H1}	I_3	I_4	I_{H2}	I_5	I_6	I_{H3}	I_7	I_8	I_{H4}
20	30	10	40	50	20	60	70	30	80	90	40

Çizelge 2.9 : Zayıf evirtimde çalışan ÇD-2 yapısı ile gerçekleştirilen 2-D sınıflandırıcı yapısı kontrol akımları (akımlar nA).

Çekirdek Devre-1	I_1 (nA)	I_2 (nA)	I_H (nA)	Güç Tüketimi
I	70	140	60	0.86 μ W
II	210	280	120	1.83 μ W
III	60	140	100	1.18 μ W
IV	280	360	80	1.74 μ W

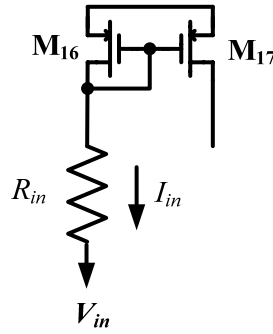
Şekil 2.19'da gösterildiği gibi I_{in1} ve I_{in2} girişleri iki boyutlu giriş verilerini göstermektedir. İki boyutlu sınıflandırıcının $(I_{in1}-I_{in2})-I_{out}$ giriş-çıkış karakteristiği benzetim sonucu Şekil 2.42'de verilmiştir. Bu şekilden görüldüğü gibi $I_{out}=0$ bölgesi de dahil edildiği zaman 9 farklı veri sınıflandırılabilir.



Şekil 2.42 : Zayıf evirtimde çalışan ÇD-2 yapısı ile gerçekleştirilen 2-D sınıflandırıcı $(I_{in1}-I_{in2})-I_{out}$ giriş-çıkış karakteristiği.

2.2.2.2 Çekirdek devre-2'nin ayrık elemanlar ile gerçekleştirilmesi

Şekil 2.30'daki devre CD4007 CMOS tranzistor entegresi kullanılarak kurulmuştur. $V_{DD}=5$ V, $V_{SS}=-5$ V seçilmiştir. Devre içindeki her bir akım kaynağı yerine Şekil 2.43'de gösterildiği gibi seri bir direnç konularak kontrol ve giriş akımları elde edilmiştir. Direnç uçlarına bağlanan gerilim kaynakları ile istenilen kontrol ve giriş akım değerleri sağlanmıştır. Akım kaynakları yerine kullanılan direnç değerleri Çizelge 2.10'da verilmiştir.

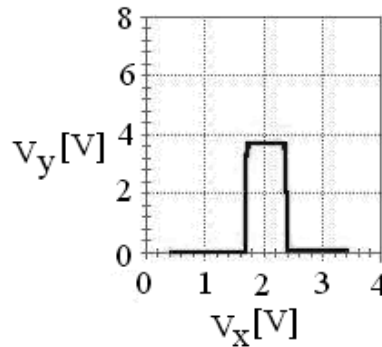


Şekil 2.43 : ÇD-2 yapısının testinde kullanılan akım kaynağı modeli.

Çizelge 2.10 : Akım kaynakları yerine kullanılan direnç değerleri.

I_{in}	I_1	I_2	I_H
$R_{in}=20\text{ k}\Omega$	$R_1=20\text{ k}\Omega$	$R_2=20\text{ k}\Omega$	$R_H=39\text{ k}\Omega$

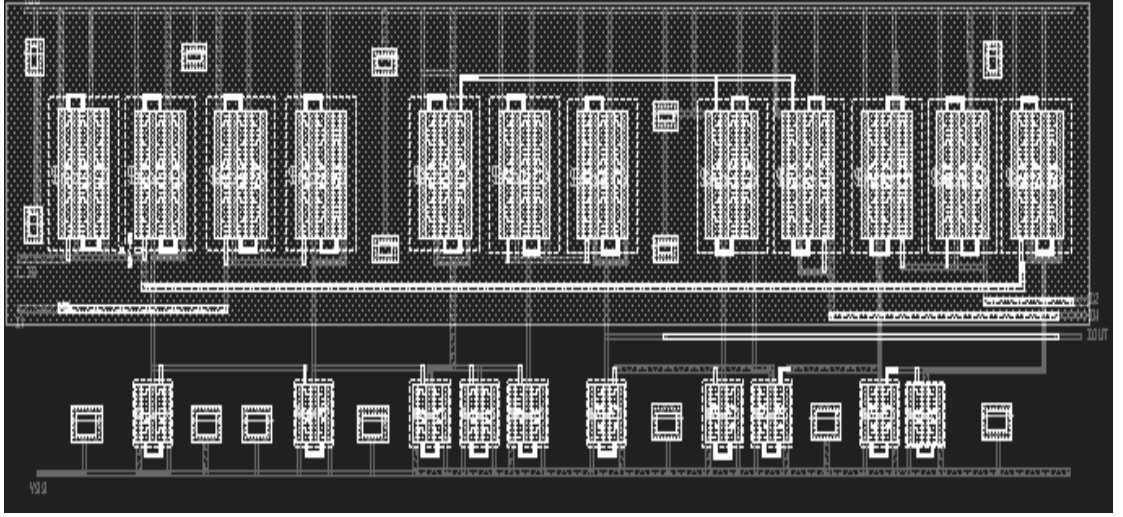
Çekirdek devre-2'yi sınamak için, $V_{R1}=1.7$ V, dolayısıyla $I_1=85\text{ }\mu\text{A}$, $V_{R2}=2.4$ V, dolayısıyla $I_2=120\text{ }\mu\text{A}$ ve $I_H=40\text{ }\mu\text{A}$ olacak şekilde ayarlanmıştır; çıkışta $100\text{ k}\Omega$ 'luk direnç bağlanmıştır. Bu direnç, akımın değişimini osiloskopta gözleyerek akımı gerilime dönüştürmek için kullanılmıştır. Kurulan devrenin giriş-çıkış karakteristiğinin test sonuçları osiloskobun X-Y özelliği kullanılarak Şekil 2.44'de verilmiştir; osiloskobun X eksenı skalası 1 V/div , Y eksenı içinse 2 V/div seçilmiştir.



Şekil 2.44 : ÇD-2 giriş-çıkış karakteristiği osiloskop sonucu.

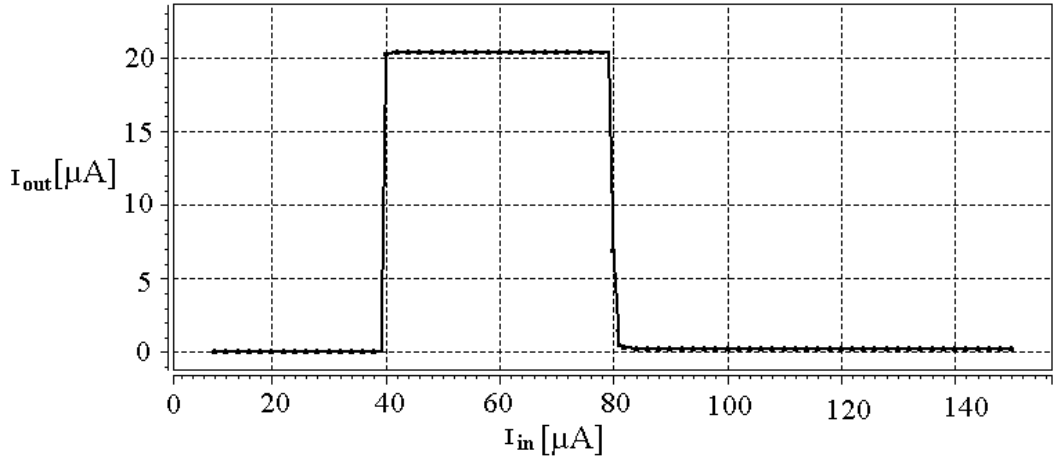
2.2.2.3 Çekirdek devre-2'nin serimi

Devre şeması Şekil 2.30'da verilmiş olan yapının serimi MENTOR programı kullanılarak 0.35 μm AMS CMOS teknoloji parametreleriyle Şekil 2.45'deki biçimde tasarlanmıştır. Serimin gerçekleştirildiği kırmık alanı 756 μm^2 olarak elde edilmiştir.



Şekil 2.45 : ÇD-2'nin serim çizimi.

Çekirdek devre-2 yapısının serim sonrası benzetimi $I_1=40 \mu\text{A}$, $I_2=80 \mu\text{A}$ ve $I_H=20 \mu\text{A}$ için Şekil 2.46'da verilmiştir.



Şekil 2.46 : ÇD-2'nin serim sonrası giriş-çıkış karakteristiği.

Bu şekilden de görüldüğü gibi çekirdek devre-2'nin Şekil 2.32'de verilen SPICE benzetimi ile serim sonrası benzetiminin giriş-çıkış karakteristiği uyuşmaktadır.

2.2.3 Çekirdek devrelerin karşılaştırılması

Çizelge 2.11’de, tezde tasarlanmış olan sınıflandırıcı devrelerin literatürde bulunan benzer özellikler taşıyan devreler ile karşılaştırılması özetlenmiştir. Çizelge 2.11’de verilmiş olan karşılaştırmada [54] numaralı kaynaktaki devrenin, $I_1=40 \mu A$, $I_2=80 \mu A$ ve $I_H=20 \mu A$ seçilmesi durumunda güç tüketimi 0.12 mW ve yayılma gecikmesinin 5 ns seviyesinde elde edildiği görülürken aynı akım değerleri için [71] numaralı kaynaktaki verilen devrenin güç tüketimi 0.38 mW olurken yayılma gecikmesi 4 ns olmuştur. Çekirdek devre-2 yapısındaki tranzistorların zayıf evrimde çalışması sayesinde güç tüketimi 0.63 μW ve yayılma gecikmesi de 1.8 μs olarak elde edilmiştir [58].

Çizelge 2.11 : Sınıflandırıcı devrelerin karşılaştırılması.

Kaynak	Üretim Teknolojisi	Besleme Gerilimi	Güç Tüketimi	Yayılma Gecikmesi	Açıklama
[1]	0.6 μm CMOS	3.3 V	14.95 mW	-	-
[47]	0.5 μm CMOS	3.3 V	90 μW -160 μW	20 μs -40 μs	-
[59]	1.2 μm CMOS	3 V	25 mW	-	-
[51]	0.35 μm CMOS	5 V	1.25 mW	-	-
[60]	0.5 μm CMOS	± 2.5 V	10 mW	0.4 μs -0.6 μs	-
[61]	0.35 μm CMOS	± 2 V	2.5 mW	-	-
[36]	0.35 μm CMOS	5 V	-	-	-
[39]	0.5 μm CMOS	4 V	80 nW-840 nW	-	-
[54]	0.35 μm CMOS	± 1.25 V	0.12 mW	5 ns	ÇD-1, $I_1=40 \mu A$, $I_2=80 \mu A$ $I_H=20 \mu A$
[58]	0.35 μm CMOS	± 1.65 V	0.63 μW	1.8 μs	ÇD-2, $I_1=50$ nA, $I_2=100$ nA $I_H=40$ nA
[55]	0.35 μm CMOS	± 1.65 V	91 μW	28 ns	ÇD-2, $I_1=8 \mu A$, $I_2=16 \mu A$, $I_H=4 \mu A$
[55]	0.35 μm CMOS	± 1.65 V	0.38 mW	4 ns	ÇD-2, $I_1=40 \mu A$, $I_2=80 \mu A$, $I_H=20 \mu A$

Çekirdek devrelerinin sınıflandırıcı olarak kullanılacağı düşünüldüğünden, öncelikli olarak kullanılacak uygulamada istenilen çalışma değerleri göz önüne alınarak yapı seçilmelidir. Bunun için önerilen devrelerde güç tüketimi, besleme gerilimi ve yayılma gecikmeleri incelenmiştir. Her ne kadar devrenin yayılma gecikmesinden

feragat edilmiş olsa bile, güç tüketimi konusunda bir iyileşme sağlanmıştır. Dolayısıyla sınıflandırma için önem taşıyan özellikler iyileştirildiği gibi, ayrıca sınıflandırma uygulamasına göre bu sakıncalar ve üstünlükler göz önüne alınarak uygun devre de seçilebilir. Ancak her iki çekirdek devre topolojisi de önerilmiş olan uygulama alanlarında kullanılabilir.

Tez çalışmasının devamında çekirdek devre olarak çekirdek devre-2 yapısı seçilmiştir. Gerek uygulamalarda, gerekse algoritma anlatımlarında kullanılan ve şekillerde çekirdek devre olarak gösterilen blok çekirdek devre-2 yapısıdır.

2.3 Çekirdek Devreler ile Elde Edilebilen Bölgeler

Tasarlanmış olan iki adet çekirdek devre yapısı da uygun kontrol akımları seçilerek sınıflandırıcı yapısı olarak kullanılabilir. Diğer taraftan, bu aşamaya kadar gerçekleştirilen çekirdek devreler kullanılarak geliştirilen sınıflandırıcı devreleri, Şekil 2.6 ile gösterildiği gibi, dikdörtgen ızgaralarla (DI) ayrıştırılabilen verileri sınıflama yeteneğine sahiptir.

Ayrıca çekirdek devre yapıları kullanılarak Şekil 2.5'deki yapı üç farklı şekilde genelleştirilebilir:

- a) m tane çekirdek bloğa x_1 değişkeni ve n tane çekirdek bloğa x_2 değişkeni uygulandıktan sonra tüm çıkışlar paralel bağlanır. Bu durumda elde edilen sınıflandırma yapısı iki boyutlu $(m+1) \times (n+1)$ tane farklı veriyi sınıflandırır. Bu şekildeki bir topolojinin veri tanımasını yapabilmesi için çıkışının ne değerler alması gerektiği Şekil 2.47'de gösterilmiştir. Blokların içinde yazılı değerler sınıflandırılacak verinin ait olduğu sınıfın seviyesini (temel çekirdek yapısı için k parametresi) göstermektedir.
- b) n tane çekirdek bloğa farklı giriş değişkenleri uygulandıktan sonra çıkışlarını paralel bağlayarak $n \times 1$ -boyutlu bir sınıflandırma yapısı elde edilebilir.
- c) (a) ve (b) maddelerindeki durumların kombinasyonunu kullanarak ve çeşitli çıkışları da aralarında gruplayarak çıkışı da çok boyutlu (bir vektör olan), farklı karar bölgeleri elde edilebilir.

		1	2	...	n
		$2k+(m+1)\Delta k$	$2(k+m\Delta k)+2\Delta k$	$\dots$	$n(k+m\Delta k)+n\Delta k$
1	$k+\Delta k$	$2k+(m+2)\Delta k$	$3k+(2m+3)\Delta k$		$(n+1)k+[n(m+1)+1]\Delta k$
2	$k+2\Delta k$	$2k+(m+3)\Delta k$	$3k+(2m+4)\Delta k$		$(n+1)k+[n(m+1)+2]\Delta k$
	$\vdots$				
m	$k+m\Delta k$	$2k+(2m+1)\Delta k$	$3k+(3m+2)\Delta k$		$(n+1)k+[n(m+1)+m]\Delta k$

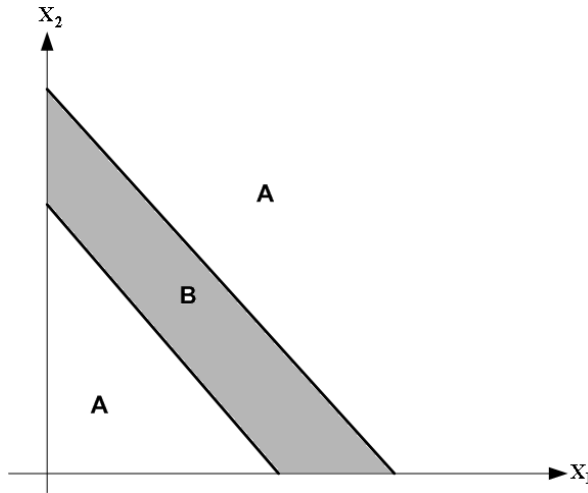
Şekil 2.47 : 2×1 boyutlu sınıflandırıcının farklı çıkış seviyeleri için genel hali.

Çalışmanın devamında yukarıda bahsedilen üç farklı durum göz önüne alınarak sınıflandırıcı çekirdek devre blokları çeşitli uygulama alanlarında kullanılmış ve benzetimleri yapılmıştır.

3. EĞİK IZGARALI SINIFLANDIRICILAR

3.1 Bölgelerin Oluşturulması

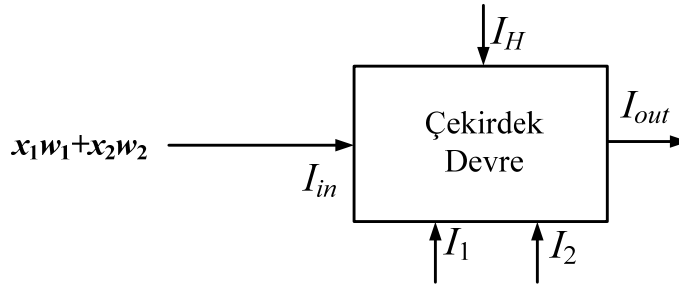
Literatürde, “perseptron” türü yapay sinir ağlarının sınıflandırabileceği verilerin ne türden olacağı araştırılırken bunların lineer olarak ayrılabilen (bölgelerin bir hiperdüzlemle ayrıştırılabildiği özel durum) veriler olduğu gösterilmiştir [19,62]. Ancak bir çok veri türünün lineer olarak ayrıştırılamadığı bilindiğinden, çalışmanın bu kısmında daha genel dağılımı olan verilerin sınıflandırıcı devreler ile nasıl ayırt edilebileceği ele alınmıştır. Lineer olarak ayrıştırılamayan verilerin sınıflandırılması için önerilmiş olan bir yöntem ise verilerin Şekil 3.1’de gösterildiği biçimde taranmış bölgeler ile ayrıştırılmasına dayanmaktadır [63].



Şekil 3.1 : Lineer olarak sınıflandırılamayan veri kümesi.

Anlatım kolaylığı sağlaması açısından, iki boyutlu iki farklı sınıftan oluşan ve lineer olarak ayrıştırılamayan Şekil 3.1’deki gibi bir veri kümesi ele alalım; bu türden bir veri yapısı için bölgeleri ayıran eğrilerin bir hiperdüzlem (2-D durumunda eğimi sonlu ve sıfırdan farklı bir doğru) olması gerekir. Eşdeğer bir başka yaklaşım ise çekirdek devrenin girişine sınıflandırılacak verilerin (x_1, x_2) belli katsayılarla ile çarpılmış (ağırlıklı) toplamını (lineer kombinasyonu) uygulamaktır.

Ağırlık katsayıları ve çekirdek devrenin kontrol akımları uygun bir sınıflandırma algoritması ile belirlenerek, Şekil 3.3’de gösterilen bölgelerin oluşması sağlanır.

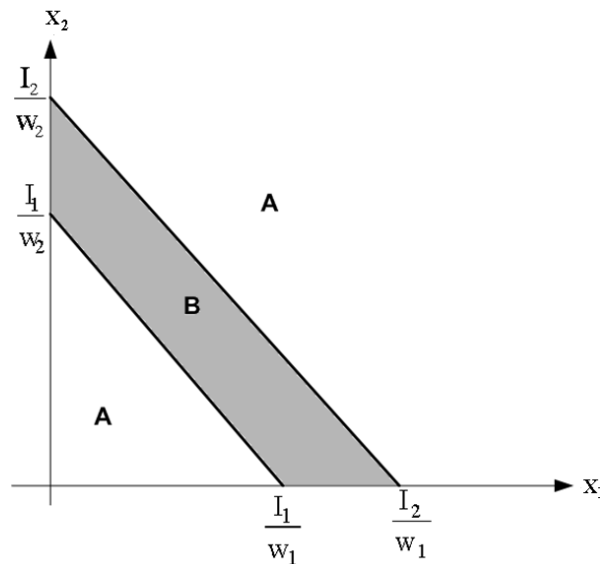


Şekil 3.2 : Eğik Izgaralı veri sınıflandırıcısı.

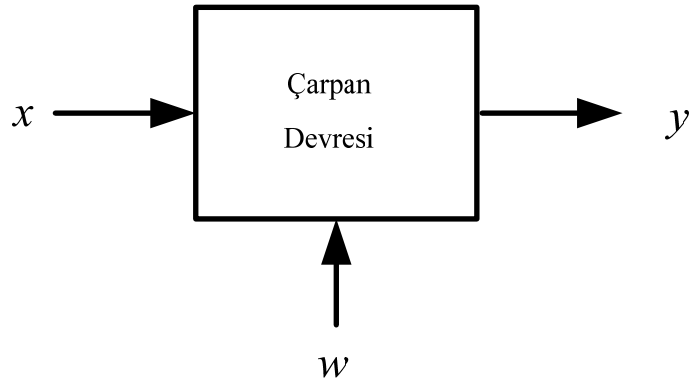
Şekil 3.2’de gösterilen yapıda I_{in} girişinin elde edilmesinde kullanılan x_1 girişinin belli bir katsayı ile çarpılmasını sağlayan Çarpan Devresi (ÇAD) yapısı Şekil 3.4’de ve tanım bağıntısı da,

$$y = xw \quad (3.1)$$

olarak verilir. Bu bağıntıdaki w bir sabit olup, giriş büyüklüğünü sadece ölçeklemektedir. Gerçekleştirilmiş olan çekirdek devre yapıları akım-modlu olduğu için (3.1) ifadesindeki xw terimi de akım modludur. Buradaki x terimi sınıflandırılacak verinin özelliklerini belirleyen bir değişkeni göstermektedir.

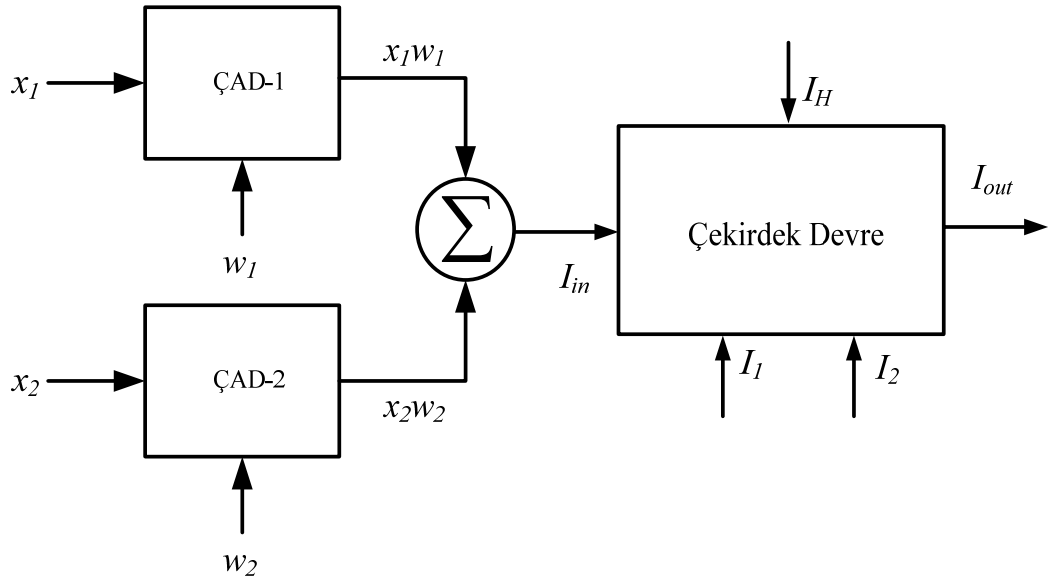


Şekil 3.3 : Dikdörtgen Izgara olmayan veri bölgeleri.



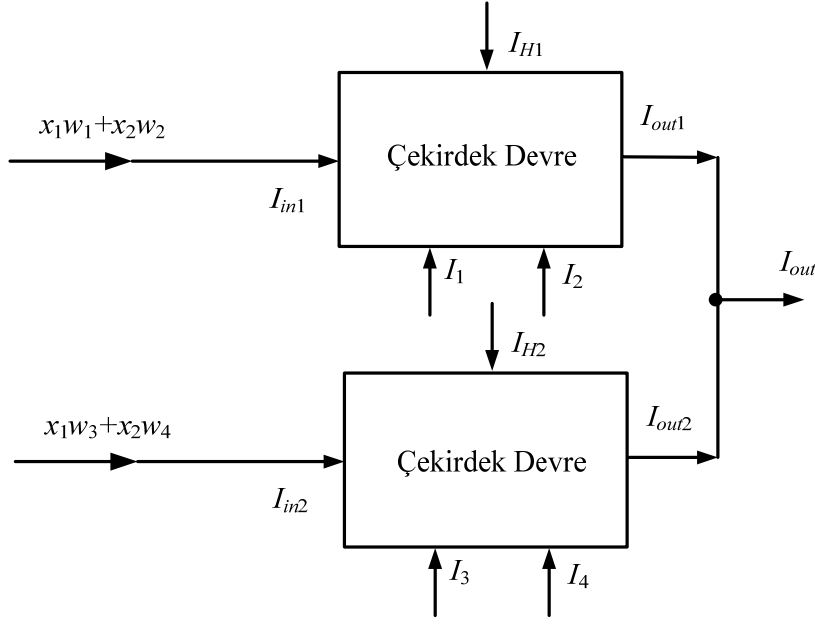
Şekil 3.4 : ÇAD yapısı blok diyagramı sembolik gösterimi.

Şekil 3.2'deki bloğun ÇAD yapıları kullanılarak elde edilen blok diyagramı Şekil 3.5'de gösterilmiştir.

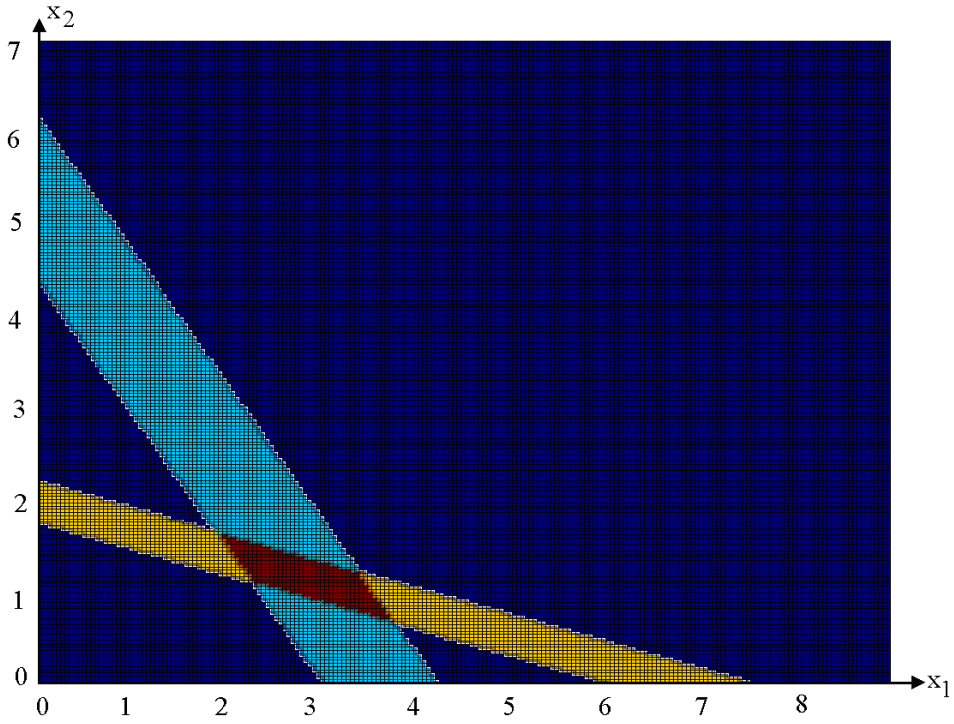


Şekil 3.5 : Şekil 3.2'deki bloğun iç yapısı.

Şekil 3.2'deki yapılar paralel bağlanarak Şekil 3.6'daki eğik ızgaralı sınıflandırıcı elde edilir; eğik ızgaralı sınıflandırıcının ayırdığı iki boyutlu veri bölgelerinin gösterilimi Şekil 3.7'de, bu bölgelerdeki verileri işleyen sınıflandırıcı çıkışının üç boyutlu $I_{out}-(x_1-x_2)$ karakteristiği ise Şekil 3.8'de verilmiştir. Buradaki üç boyutlu şekilde yükseklik değeri çekirdek devrelerin I_H parametresi ile ayarlanmaktadır. Diğer bir ifade ile, sınıflandırılacak veriler bu yüksekliklerin aldığı değerlere göre saptanmaktadır. Dolayısıyla tipik birer eğri örneği olan Şekil 3.7'de ve Şekil 3.8'de her renk farklı bir veri sınıfına karşı gelmektedir.

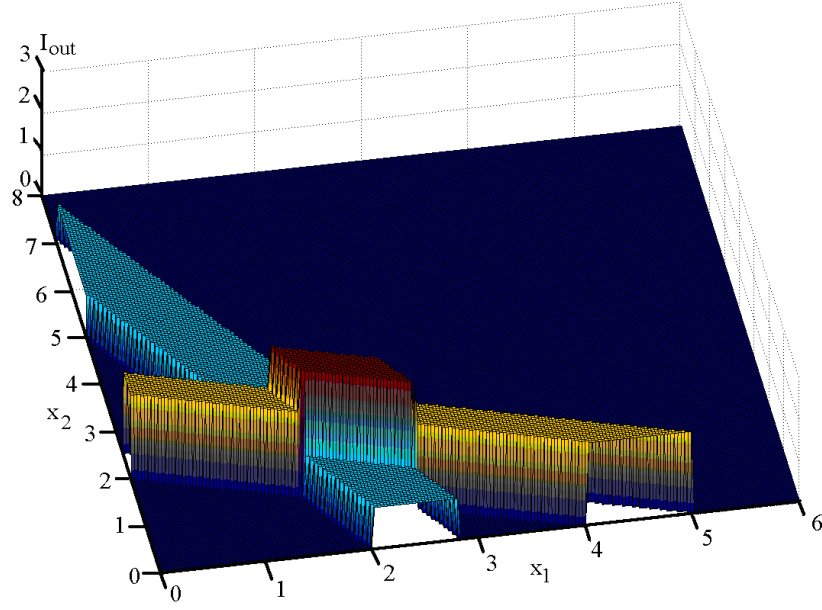


Şekil 3.6 : Paralel bağlanmış sınıflandırıcı devresi.



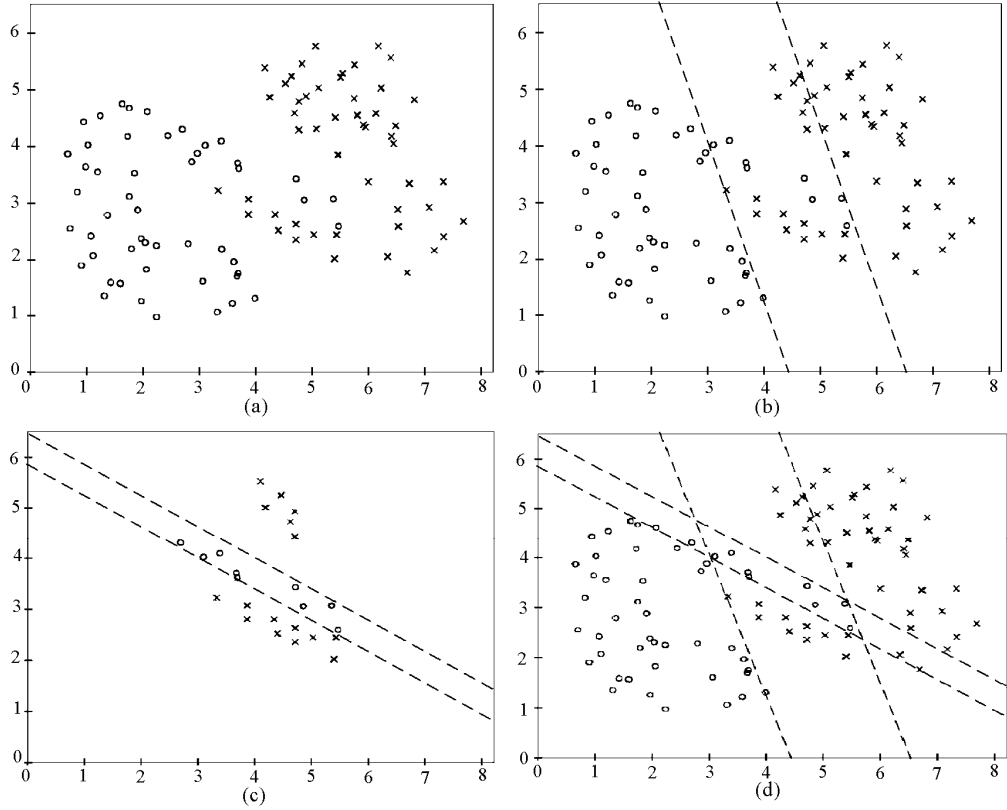
Şekil 3.7 : Eğik ızgaralı veri sınıfları.

Şekil 3.2'deki yapılardan ikiden fazla sayıda paralel yapı bağlanarak ya da girişlerine 2 veri değişkeninden daha fazlasının lineer kombinezonu uygulanarak, daha farklı ve çok sayıda bölgeler de oluşturulabilir. ÇAD devresinin donanımsal gerçekleştirilmesi Bölüm 3.2'de incelenmiştir.



Şekil 3.8 : Eğik ızgaralı sınıflandırıcı çıkışının 3-D $I_{out}(x_1-x_2)$ karakteristiği.

Şekil 3.9’da gösterilen, lineer olarak sınıflandırılmayan ve de dikdörtgen ızgara bölgeler ile ayrıştırılmayan veriler, ÇAD ve ÇD yapıları kullanılarak sınıflandırılabilir. Bu verilerin sınıflandırılması biçimsel olarak aşama aşama Şekil 3.9 a-d’de gösterilmiştir.



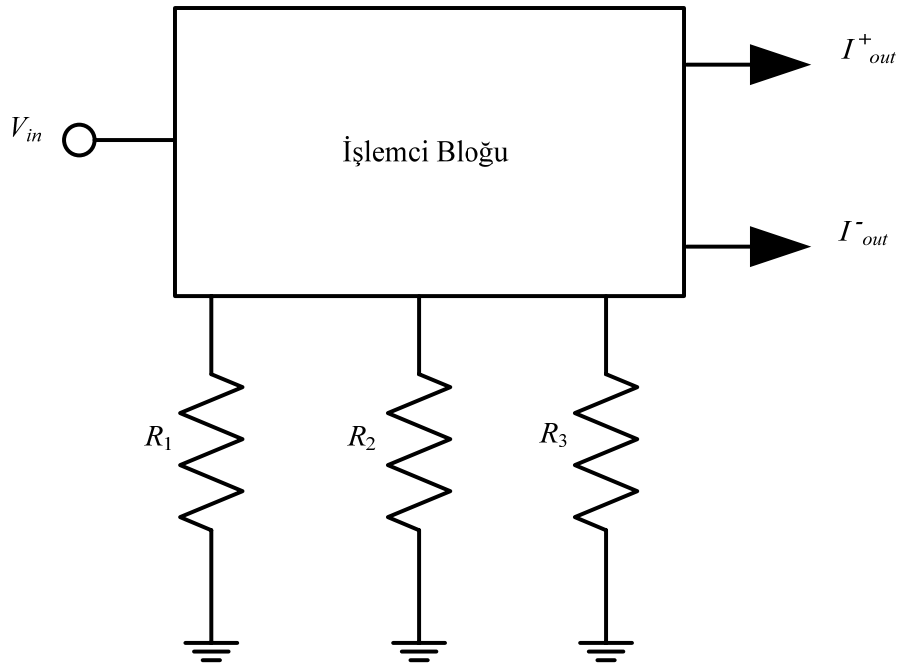
Şekil 3.9 : Verilerin ÇAD ve ÇD kullanılarak sınıflandırılması.

Dolayısıyla bu çeşit bir veri kümesinin sınıflandırılabilmesi için uygun doğruların, başka bir deyişle ÇAD devresinin parametreleri olan w_i 'ler (ağırlık katsayıları) ve ÇD parametreleri I_1 , I_2 (kontrol akımları) değerlerinin bulunması gerekmektedir. Şekil 3.9'da kesikli çizgiler ile gösterilen bu doğrulara çift eşik doğruları denilmektedir.

Bu çalışmada çift eşik doğrularının elde edilmesi için Fisher'in lineer diskriminant analiz yönteminden faydalanılmıştır [64]. Şekil 3.8'de gösterilen yüksekliklerin sayısal değerleri ÇD parametresi I_H ile ayarlanmaktadır. Dolayısıyla her farklı yükseklik değeri ayrı bir sınıfı göstermektedir. Şekil 3.6'da gösterilmiş bir sınıflandırıcı yapısı uygulaması Bölüm 5.3'de verilmiştir.

3.2 ÇAD Devresi ve Benzetimleri

Çarpan devresinin blok diyagramı Şekil 3.4'te gösterilmiştir. Tanım bağıntısı (3.1) ifadesi ile verilen bu yapı, uygulamalarda kullanım kolaylığı sağlanması için giriş değişkeni gerilim, çıkış değişkeni akım olarak tasarlanmıştır. Dolayısıyla gerçekleştirilmiş olan devre blok şeması olarak Şekil 3.10'da gösterilmiştir.



Şekil 3.10 : ÇAD devresi blok şeması.

Bu devrenin tanım bağıntıları da

$$I_{out}^+ = \frac{V_{in}}{R_1} \frac{R_2}{R_3} \quad (3.2)$$

$$I_{out}^- = -\frac{V_{in}}{R_1} \frac{R_2}{R_3} \quad (3.3)$$

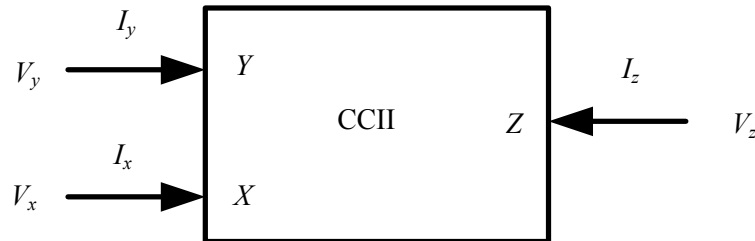
ifadeleri ile verilmiştir.

Şekil 3.10’da kullanılan R_1 direnci, V_{in} giriş geriliminin akıma dönüştürülmesini sağlamak amacıyla kullanılmıştır. Ayrıca, aynı direnç uygulamalarda karşılaşılabilecek giriş verilerinin uygun değerlere normalizasyonu için de kullanılmaktadır. Diğer taraftan R_2/R_3 oranı ise giriş geriliminin, dolayısıyla akımının belli bir çarpan ile çıkışa aktarılmasını sağlamaktadır. Burada kullanılan elemanların direnç olması gerek çarpım gerekse normalizasyon işlemi sırasında esneklik sağlamaktadır. Ayrıca (3.2) ve (3.3) ifadelerinden görüldüğü gibi çıkış değişkeni, akım boyutunda oluşturulduğundan çekirdek devre yapılarına uygulanabilmektedir.

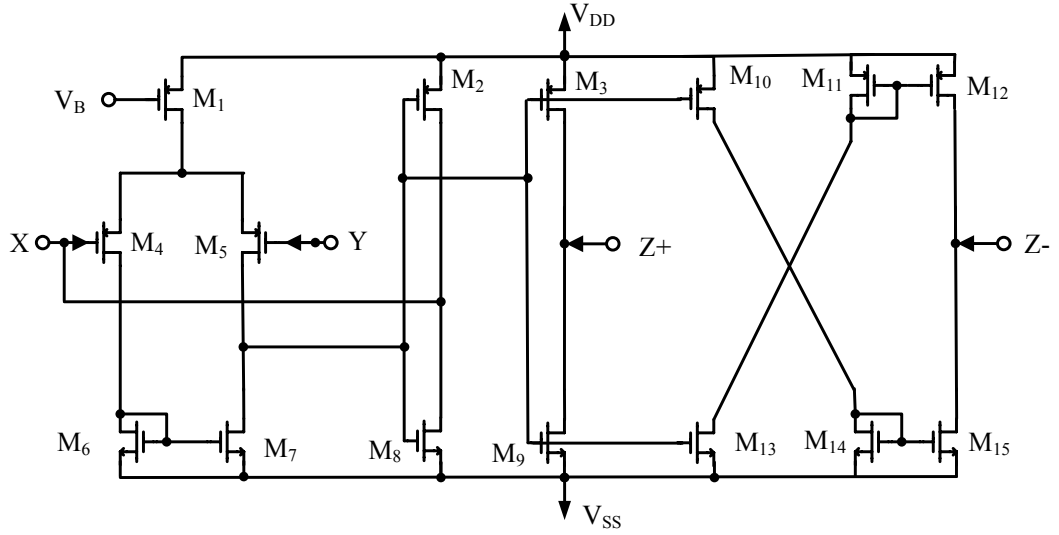
ÇAD devresinin devre şeması iki adet İkinci Kuşak Akım Taşıyıcı (CCII) elemanı kullanılarak gerçekleştirilmiştir [65]. Bu CCII yapılarında z ucu çıkış kabul edilmek üzere, tanım bağıntısı aşağıdaki gibi

$$\begin{bmatrix} V_x \\ I_y \\ I_z \end{bmatrix} = \begin{bmatrix} 0 & \beta & 0 \\ 0 & 0 & 0 \\ \alpha & 0 & 0 \end{bmatrix} \begin{bmatrix} I_x \\ V_y \\ V_z \end{bmatrix} \quad (3.4)$$

matris biçiminde ifade edilen CCII yapısının blok diyagramı Şekil 3.11’de ve devre şeması da Şekil 3.12’de gösterilmiştir.

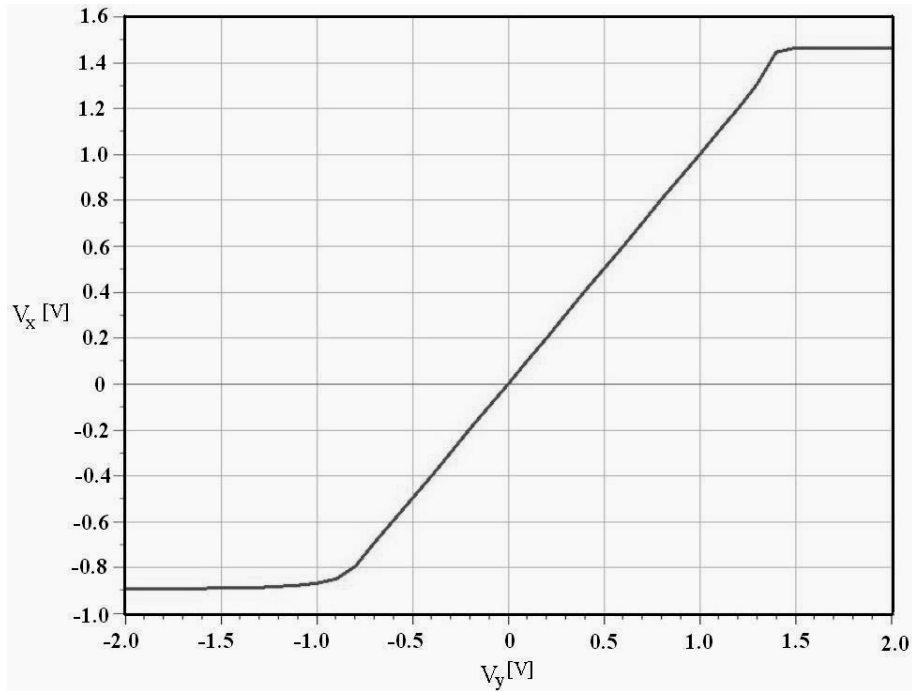


Şekil 3.11 : CCII devresi blok diyagramı.



Şekil 3.14 : DO-CCII devre şeması.

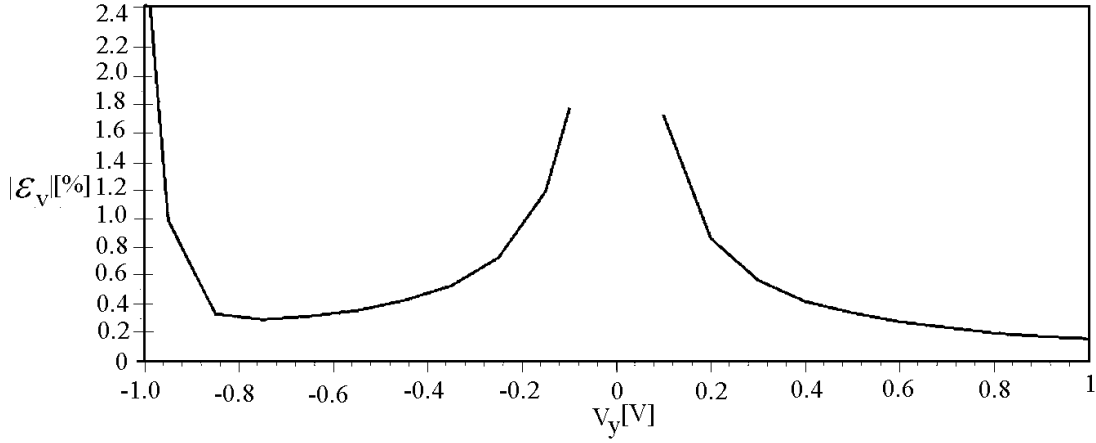
DO-CCII devresinin benzetiminde 0.35 μm AMS CMOS teknoloji parametreleri ve besleme gerilimleri de V_{DD} ve $V_{SS} \pm 1.65$ V olarak kullanılmıştır. Kutuplama akımının 18 μA olması için V_B gerilimi 0.8 V olarak seçilmiştir. Şekil 3.14’de verilen devre için V_x geriliminin V_y ile değişim karakteristiği Şekil 3.15’de verilmiştir.



Şekil 3.15 : DO-CCII devresi için V_x 'in V_y ile değişim karakteristiği.

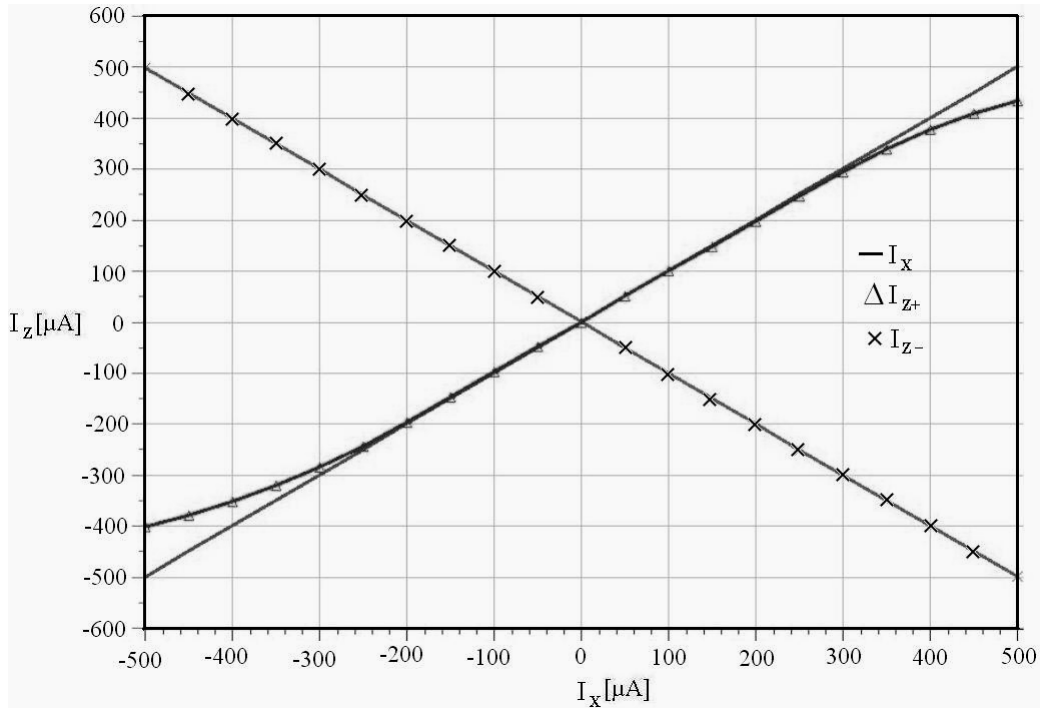
İdeal olarak DO-CCII devresinde $\alpha_1 = \alpha_2 = 1$ ve $\beta = 1$ 'dir. Ancak devredeki idealsizliklerden dolayı $\beta = 1 - \varepsilon_v$, $\alpha_1 = 1 - \varepsilon_{i1}$ ve $\alpha_2 = 1 - \varepsilon_{i2}$ değerleriyle verilebilir. Buradaki ε_v ve ε_{ik} ($k=1,2$; $|\varepsilon_v| \ll 1$; $|\varepsilon_{ik}| \ll 1$) sayıları sırası ile gerilim ve akım

izleme hatalarını vermektedir. Gerilim izleme hatasının incelenmesi için $|\varepsilon_v|$ 'nin V_y ile değişim karakteristiği Şekil 3.16'da verilmiştir. Şekilden görüldüğü gibi gerilim izleme hatası maksimum % 2.4 olmaktadır. ε_v 'nin değişimi 0 noktasında tanımsız olmasından dolayı bu nokta civarı için karakteristik çizdirilememiştir.



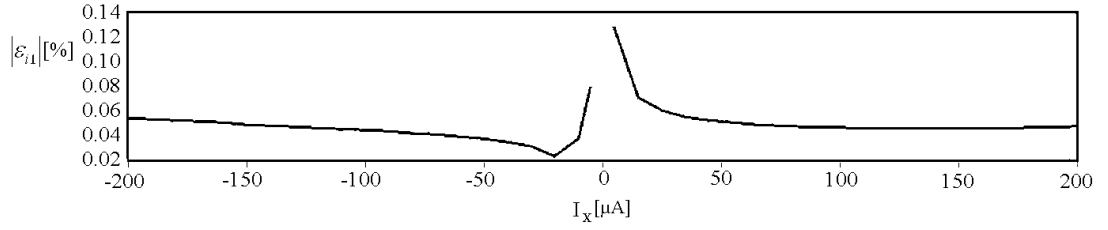
Şekil 3.16 : DO-CCII devresi için $|\varepsilon_v|$ 'nin V_y ile değişim karakteristiği.

Ayrıca aynı devrede I_{z+} ve I_{z-} akımlarının I_x akımını takip etme başarımlarının gösterildiği karakteristik ise Şekil 3.17'de verilmiştir.

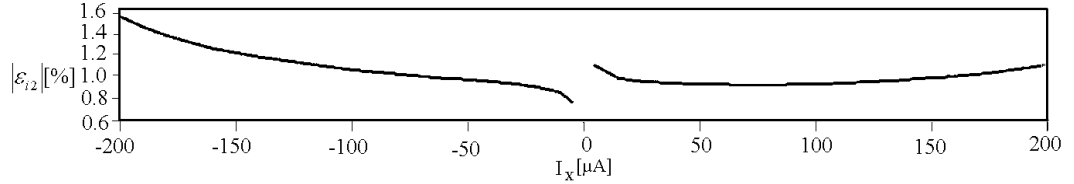


Şekil 3.17 : DO-CCII için I_{z+} ve I_{z-} akımlarının I_x ile değişim karakteristiği.

Benzer şekilde akım izleme hatalarının incelenmesi için $|\varepsilon_{i1}|$ ve $|\varepsilon_{i2}|$ 'nin I_x ile değişim karakteristikleri sırası ile Şekil 3.18'de ve Şekil 3.19'da verilmiştir.



Şekil 3.18 : DO-CCII devresi için $|\varepsilon_{11}|$ 'nin I_x ile değişim karakteristiği.



Şekil 3.19 : DO-CCII devresi için $|\varepsilon_{12}|$ 'nin I_x ile değişim karakteristiği.

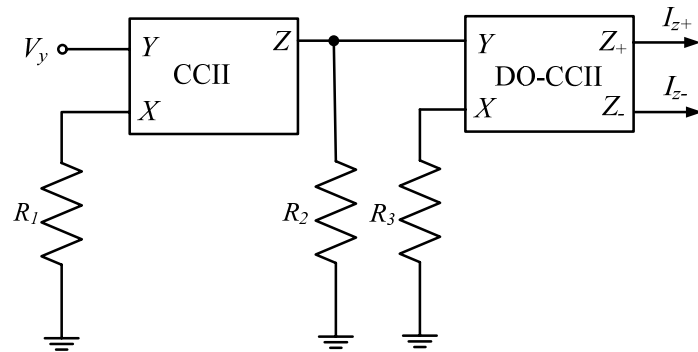
Karakteristiklerden görüldüğü gibi giriş akımı $-200 \mu A$ ile $200 \mu A$ aralığında değiştirildiği zaman, maksimum akım izleme hatası $|\varepsilon_{11}| = \% 0.13$ ve $|\varepsilon_{12}| = \% 1.6$ 'dır.

Çalışmada sınıflandırma için kullanılacak temel yapılardan biri olan ÇAD devresi, Şekil 3.11 ve Şekil 3.13'deki yapılar kullanılarak Şekil 3.20'de gösterildiği biçimde oluşturulmuştur. Bu yapının akım gerilim bağıntısı aşağıdaki ifadeler ile verilir:

$$I_{z+} = \frac{V_y R_2}{R_1 R_3} \quad (3.6)$$

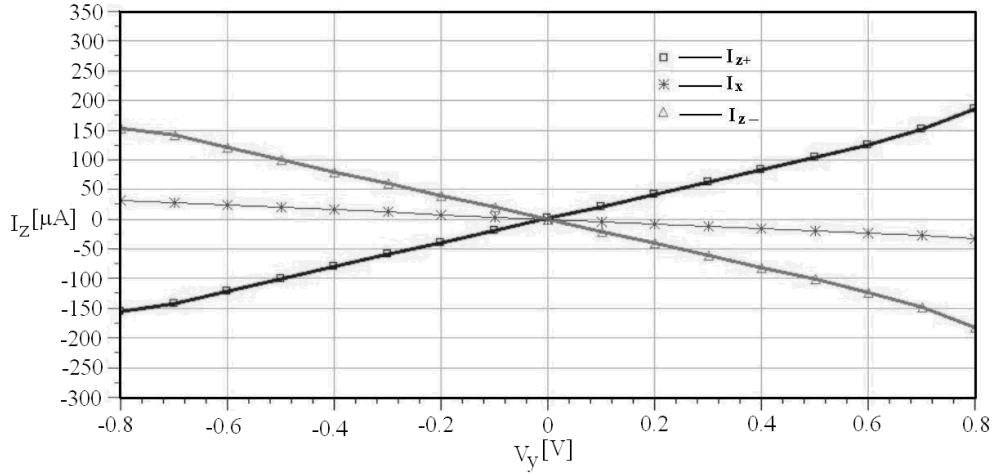
$$I_{z-} = -\frac{V_y R_2}{R_1 R_3} \quad (3.7)$$

Çıkış değişkeni olarak (3.6) ve (3.7) ifadesindeki I_{z+} ve I_{z-} akımları kullanılmış ve uygulamalarda (3.2) ifadesinde gösterildiği şekilde I_{out} olarak alınmıştır.



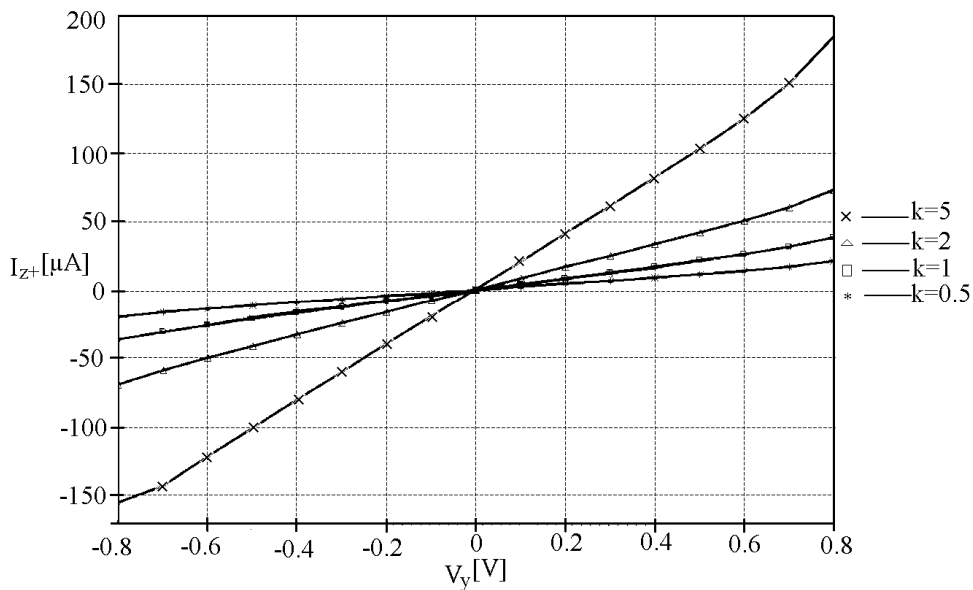
Şekil 3.20 : ÇAD yapısının CCII ve DO-CCII yapıları ile gerçekleştirilmesi.

Şekil 3.20’de verilen devrenin benzetimi için $R_1=25\text{ k}\Omega$, $R_2=25\text{ k}\Omega$ ve $R_3=5\text{ k}\Omega$ seçilerek V_y ucuna -0.8 V ile 0.8 V arasında değişen gerilim uygulanmış ve R_1 direnci üzerindeki akım giriş akımı olarak alınmıştır. Dolayısıyla giriş akımı $-32\text{ }\mu\text{A}$ ile $32\text{ }\mu\text{A}$ aralığında değiştirilmiştir. $R_2 / R_3 = 5$ için I_{z+} ve I_{z-} akımlarının V_y gerilimi ile değişim karakteristiği benzetim sonucu Şekil 3.21’de verilmiştir.



Şekil 3.21 : ÇAD devresi için I_{z+} ve I_{z-} akımlarının V_y gerilimi ile değişim karakteristiği ($R_2 / R_3 = 5$).

ÇAD devresinde R_2 / R_3 oranlarının farklı seçilmesi durumunda giriş-çıkış karakteristiği için gerçekleştirilmiş olan benzetim sonucu Şekil 3.22’de verilmiştir. Bu benzetimde $R_2=25\text{ k}\Omega$ alınmış ve R_3 değerleri şekildeki k değerlerini sağlayacak biçimde seçilmiştir.



Şekil 3.22 : Çeşitli $k = R_2 / R_3$ değerleri için ÇAD devresi $V_y - I_{z+}$ karakteristiği.

Önerilen ÇAD devresinin benzetiminde kullanılan tranzistor boyutları Çizelge 3.1’de verilmiştir.

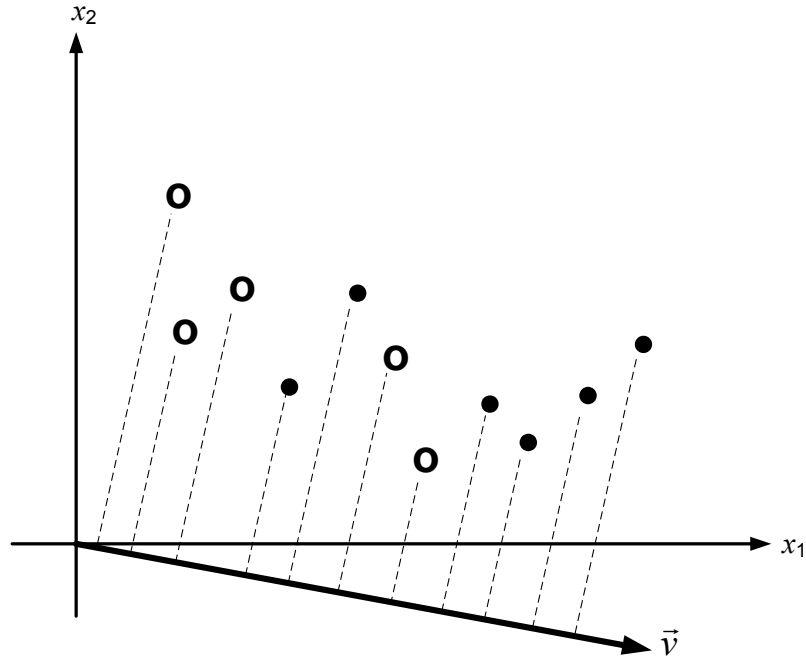
Çizelge 3.1 : ÇAD devresi MOS tranzistorların boyutları.

MOSFET	W [μm]	L [μm]
M ₁ , M ₄ , M ₅	21	1.05
M ₆ , M ₇	6.3	1.05
M ₂ , M ₃ , M ₁₀ , M ₁₁ , M ₁₂	42.7	1.05
M ₈ , M ₉ , M ₁₃ , M ₁₄ , M ₁₅	11.2	1.05

4. SINIFLANDIRMA ALGORİTMALARI VE DEVRELERE UYGULANMASI

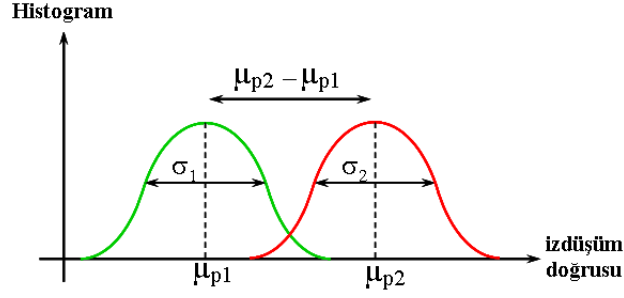
4.1 Fisher Tabanlı Algoritma ile Çift Eşik Doğrularının Bulunmalarına Genel Bakış

Sınıflandırılacak veriler 2 boyutlu uzayda, c_1 ve c_2 sınıflarına ait olmak üzere, Şekil 4.1’de gösterildiği gibi verilmiş olsun. Fisher’in lineer diskriminant analizi geliştirilerek bu verilerin, $\vec{v}$ vektörü ile doğrultusu verilen ve orijinden geçen bir doğru üzerine izdüşürülmelerinden yararlanarak ayrıştırılmasına çalışılacaktır [64].



Şekil 4.1 : Lineer olarak sınıflandırılmayan örnek veri kümesi.

Bu izdüşüm doğrusu üzerine izdüşürülen verilerin merkeze (orijine) uzaklıkları göz önüne alınarak Şekil 4.2’de gösterildiği gibi histogramları çizilir. Şekil 4.2’de μ_{p1} ve μ_{p2} verilerin orijine olan uzaklıklarının ortalamasını, σ_1 ve σ_2 ise standart sapmalarını göstermektedir.



Şekil 4.2 : Histogram karakteristiği.

Histogram eğrilerinden faydalanılarak iki eşik doğrusu elde edilir. Ancak izdüşüm doğrusu, verilerin en iyi ayrıştırılmasının sağlanması açısından:

a) μ_{p1} ve μ_{p2} birbirinden maksimum uzaklıkta

b) σ_1 ve σ_2 de olabildiğince küçük

koşullarını sağlayacak şekilde seçilmelidir.

4.1.1 Fisher tabanlı algoritma ile çift eşik doğrularının bulunması

$\vec{x}$ vektörünü

$$\vec{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \quad (4.1)$$

şeklinde verilen iki boyutlu ve iki farklı sınıftan oluşan veri kümesinin bir elemanı olarak alalım; $\vec{x}$ örneklerinin, $\vec{v}$ birim vektörü ile temsil edilen izdüşüm doğrusuna izdüşümleri $\vec{v}^T \vec{x}$ iç çarpımıyla gösterilir. μ_{p1} ve μ_{p2} sırası ile birinci ve ikinci sınıfa ait verilerin orijine olan ortalama izdüşüm uzaklıkları, $\vec{\mu}_1$ ve $\vec{\mu}_2$ ise birinci sınıf ve ikinci sınıf veri değerlerinin aritmetik ortalamalarından oluşan vektörler olsun. Bu durumda

$$\mu_{p1} = \vec{v}^T \vec{\mu}_1 \quad (4.2)$$

$$\mu_{p2} = \vec{v}^T \vec{\mu}_2 \quad (4.3)$$

bağıntıları geçerlidir [64].

Ayrıca

$$y = \vec{v}^T \vec{x} \quad (4.4)$$

ifadesi orijine olan izdüşüm uzaklıklarını gösterdiğine göre, birinci ve ikinci sınıf verileri için sırası ile saçılmalar (standart sapmanın eşdeğeri):

$$s_{p1}^2 = \sum_{y_i \in C_1} (y_i - \mu_{p1})^2 \quad (4.5)$$

$$s_{p2}^2 = \sum_{y_i \in C_2} (y_i - \mu_{p2})^2 \quad (4.6)$$

ifadeleri ile verilir.

Bu ifadelerde verilen y_i 'ler eğitim kümesindeki her bir elemanın orijine olan izdüşüm uzaklığıdır.

Sınıflandırma işleminin en az hata ile gerçekleştirilebilmesi için,

$$J(\vec{v}) = \frac{(\mu_{p1} - \mu_{p2})^2}{(s_{p1} + s_{p2})^2} \quad (4.7)$$

ifadesinin maksimum kılınması gerekir; dolayısıyla öyle bir $\vec{v}$ bulunmalıdır ki $J(\vec{v})$ ifadesi maksimum olsun.

Birinci ve ikinci sınıf verilerin sınıf içi saçılma matrisleri:

$$S_1 = \sum_{\vec{x} \in C_1} (\vec{x} - \vec{\mu}_1)(\vec{x} - \vec{\mu}_1)^T \quad (4.8)$$

$$S_2 = \sum_{\vec{x} \in C_2} (\vec{x} - \vec{\mu}_2)(\vec{x} - \vec{\mu}_2)^T \quad (4.9)$$

ifadeleri ile tanımlanır ve S_W matrisi

$$S_W = S_1 + S_2 \quad (4.10)$$

olarak yazılırsa, (4.5) ifadesi ile verilen saçılma terimleri:

$$s_{p1}^2 = \sum_{y_i \in C_1} (y_i - \mu_{p1})^2 = \sum_{y_i \in C_1} (\vec{v}^T \vec{x} - \vec{v}^T \vec{\mu}_1)^2 = \sum_{y_i \in C_1} (\vec{v}^T (\vec{x} - \vec{\mu}_1))^T (\vec{v}^T (\vec{x} - \vec{\mu}_1)) \quad (4.11)$$

$$s_{p1}^2 = \sum_{y_i \in C_1} ((\vec{x} - \vec{\mu}_1)^T \vec{v}) ((\vec{x} - \vec{\mu}_1)^T \vec{v}) = \sum_{y_i \in C_1} \vec{v}^T (\vec{x} - \vec{\mu}_1) (\vec{x} - \vec{\mu}_1)^T \vec{v} = \vec{v}^T S_1 \vec{v} \quad (4.12)$$

biçiminde ifade edilebilirler.

Benzer şekilde (4.6) ifadesi de şöyle yazılabilir:

$$s_{p2}^2 = \vec{v}^T S_2 \vec{v} \quad (4.13)$$

Bu durumda saçılma matrisleri toplamı

$$s_{p1}^2 + s_{p2}^2 = \vec{v}^T S_1 \vec{v} + \vec{v}^T S_2 \vec{v} = \vec{v}^T S_w \vec{v} \quad (4.14)$$

olur.

İki sınıfın ortalamaları arasındaki ayrılabilirliğin bir ölçütü olan sınıflar arası saçılma matrisi

$$S_B = (\vec{\mu}_1 - \vec{\mu}_2)(\vec{\mu}_1 - \vec{\mu}_2)^T \quad (4.15)$$

ifadesi ile verilir.

Diğer taraftan (4.7) ifadesinin pay terimi aşağıdaki şekilde yazılır:

$$(\mu_{p1} - \mu_{p2})^2 = (\vec{v}^T \vec{\mu}_1 - \vec{v}^T \vec{\mu}_2)^2 = \vec{v}^T (\vec{\mu}_1 - \vec{\mu}_2)(\vec{\mu}_1 - \vec{\mu}_2)^T \vec{v} = \vec{v}^T S_B \vec{v} \quad (4.16)$$

Dolayısıyla (4.14) ve (4.16) eşitlikleri kullanılarak oluşturulmuş

$$J(\vec{v}) = \frac{(\mu_{p1} - \mu_{p2})^2}{(s_{p1}^2 + s_{p2}^2)} = \frac{\vec{v}^T S_B \vec{v}}{\vec{v}^T S_w \vec{v}} \quad (4.17)$$

ifadesinden görüldüğü gibi sınıflandırma işleminin en uygun şekilde yapılması sınıflar arası saçılma matrisine (S_B) ve sınıf içi saçılma matrisine (S_w) bağlı çıkmaktadır.

Böylece problem $J(\vec{v})$ ölçütünü maksimum kılan $\vec{v}$ vektörünü, yani izdüşüm doğrusunu bulmak şeklinde ifade edilebilir hale gelir. Bunun için ölçütün gradyenini sıfıra eşitleyen

$$\frac{\partial}{\partial \vec{v}} J(\vec{v}) = \frac{\left(\frac{\partial}{\partial \vec{v}} \vec{v}^T S_B \vec{v} \right) \vec{v}^T S_w \vec{v} - \left(\frac{\partial}{\partial \vec{v}} \vec{v}^T S_w \vec{v} \right) \vec{v}^T S_B \vec{v}}{(\vec{v}^T S_w \vec{v})^2} = 0 \quad (4.18)$$

ifadesi kullanılarak

$$\frac{\partial}{\partial \vec{v}} J(\vec{v}) = \frac{(2S_B \vec{v}) \vec{v}^T S_w \vec{v} - (2^T S_w \vec{v}) \vec{v}^T S_B \vec{v}}{(\vec{v}^T S_w \vec{v})^2} = 0 \quad (4.19)$$

$$\vec{v}^T S_w \vec{v} (S_B \vec{v}) - \vec{v}^T S_B \vec{v} (S_w \vec{v}) = 0 \quad (4.20)$$

$$\frac{\vec{v}^T S_w \vec{v} (S_B \vec{v})}{\vec{v}^T S_w \vec{v}} - \frac{\vec{v}^T S_B \vec{v} (S_w \vec{v})}{\vec{v}^T S_w \vec{v}} = 0 \quad (4.21)$$

ara işlemlerinden sonra

$$(S_B \vec{v}) - \frac{\vec{v}^T S_B \vec{v} (S_w \vec{v})}{\vec{v}^T S_w \vec{v}} = (S_B \vec{v}) - \lambda (S_w \vec{v}) = 0 \quad (4.22)$$

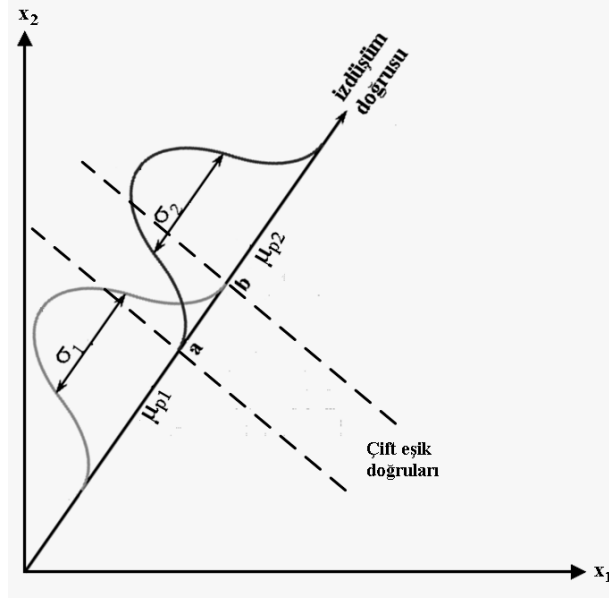
eşitliği elde edilir.

(4.22) ifadesindeki denklemin çözümü ise genelleştirilmiş özdeğer probleminden başka bir şey değildir. Bu özdeğer problemi

$$S_B \vec{v} = \lambda S_w \vec{v} \quad (4.23)$$

denklemini ile verilir.

Bu denklemin çözümünden bulunacak olan maksimum değere sahip özdeğere karşılık gelen $\vec{v}$ vektörü verilerin izdüşürüleceği doğrunun eğimini vermektedir [66]. Bu birim vektör Şekil 4.3’de gösterilen izdüşüm doğrusu ile aynı yöndedir [64].



Şekil 4.3 : Çift eşik doğrularının gösterilimi.

Şekil 4.3’de açık renkli olarak çizilmiş histogram eğrisi birinci sınıfa, koyu renkli olarak çizilmiş histogram eğrisi ikinci sınıfa ait veriler olarak alalım. Bu durumda $\vec{v}$ vektörü bulunduktan sonra histogram eğrileri kullanılarak a ve b noktaları bulunur. Burada koordinatları (x_{a1}, x_{a2}) olan a noktası, izdüşüm doğrusu üzerinde ikinci sınıfa ait olabilecek son noktadır; başka bir deyişle, izdüşüm doğrusu üzerinde ikinci sınıfa ait verilerin orijine uzaklığı en küçük olan noktadır. Benzer şekilde koordinatları (x_{b1}, x_{b2}) olan b noktası ise birinci sınıfa ait verilerde orijine olan izdüşüm uzaklığı maksimum olan değerdir. Bu durumda verilerin ayırt edilmesini sağlayacak doğruların (hiperdüzlemlerin) denklemleri ise aşağıdaki ifadeler ile verilir:

$$\vec{v}^T \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} - \vec{v}^T \begin{pmatrix} x_{a1} \\ x_{a2} \end{pmatrix} = 0 \quad (4.24)$$

$$\vec{v}^T \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} - \vec{v}^T \begin{pmatrix} x_{b1} \\ x_{b2} \end{pmatrix} = 0 \quad (4.25)$$

Bu denklemler izdüşüm doğrusu üzerinde bulunan a ve b noktalarına dik çıkan Şekil 4.3’de kesik çizgiler ile gösterilmiş doğru denklemleridir. Bu doğru denklemleri bulunduktan sonra a ile b noktaları arasında kalan veri kümesi için aynı algoritma tekrar uygulanır ve benzer doğrular arada kalan veri kümesi için tekrar hesaplanır. Verileri kesin olarak ayırıştıran doğrular bulunduktan sonra, yöntem durdurulur. Bu (4.24) ve (4.25) ifadeleri ile bulunmuş olan denklemler çift eşik doğrularıdır.

4.1.2 Fisher tabanlı algoritma ile çift eşik doğrularının bulunmasında genelleştirilmiş hal

Çift eşik doğrularının çizilmesi için anlatım kolaylığı açısından iki boyutlu ve iki sınıftan oluşan veri kümeleri kullanıldı. Çift eşik doğrularının çizilmesinde genelleştirilmiş durum, $d \geq c$ koşulu altında d boyutlu c adet farklı sınıf olsun. $c=2$ olması durumu için Altbölüm 4.1.1’de anlatılan yöntem hiç bir değişikliğe uğramaksızın uygulanır; aksi halde $(c-1)$ adet birim izdüşüm vektörü bulunur ve bunlarla

$$V = [\vec{v}_1 | \vec{v}_2 | \dots | \vec{v}_{c-1}] \quad (4.26)$$

matrisi oluşturulur.

Bu durumda izdüşüm vektörlerinin bulunması için öncelikle genelleştirilmiş haldeki sınıf içi saçılma matrisi (S_W) ve sınıflar arası saçılma matrisi (S_B) elde edilmelidir. n_i , i . sınıfa ait eğitim kümesindeki veri sayısı ise, S_W ve S_i matrisleri

$$S_W = \sum_{i=1}^c S_i \quad (4.27)$$

$$S_i = \sum_{x \in C_i} (\vec{x} - \vec{\mu}_i)(\vec{x} - \vec{\mu}_i)^T \quad (4.28)$$

ifadeleri ile, $\vec{\mu}_i$ ortalaması da

$$\vec{\mu}_i = \frac{1}{n_i} \sum_{x \in C_i} \vec{x} \quad (4.29)$$

eşitliği ile verilir. Dolayısıyla

$$S_W = \sum_{i=1}^c \sum_{C_i} (\vec{x} - \vec{\mu}_i)(\vec{x} - \vec{\mu}_i)^T \quad (4.30)$$

olur.

(4.28) ifadesinde bulunan $\vec{\mu}_i$, i . sınıfa ait verilerin ortalaması, n eğitim kümesindeki toplam veri sayısıdır. Tüm verilerin ortalaması olan $\vec{\mu}$ vektörü,

$$\vec{\mu} = \frac{1}{n} \sum_{\forall x} \vec{x} \quad (4.31)$$

ifadesi ile verilir.

Sınıflar arası saçılma matrisi S_B aşağıdaki eşitlikteki gibidir:

$$S_B = \sum_{i=1}^c n_i (\vec{\mu}_i - \vec{\mu})(\vec{\mu}_i - \vec{\mu})^T \quad (4.32)$$

$\vec{\mu}_{pi}$, i . sınıfa ait verilerin ortalama izdüşüm uzaklığından oluşan vektör, $\vec{\mu}_p$ ise $\vec{\mu}_{pi}$ 'lerin ortalamasıdır. Dolayısıyla

$$\vec{\mu}_p = \frac{1}{n} \sum_{\forall y} y \quad (4.33)$$

olarak yazılabilir.

Sınıf içi saçılma matrisi (S_{PW}) ve izdüşürülmüş sınıflar arası saçılma matrisi (S_{PB}) sırası ile

$$S_{PW} = \sum_{i=1}^c \sum_{y \in C_i} (\vec{y} - \vec{\mu}_{pi})(\vec{y} - \vec{\mu}_{pi})^T \quad (4.34)$$

olmak üzere,

$$S_{PW} = V^T S_W V \quad (4.35)$$

ile, ve

$$S_{PB} = \sum_{i=1}^c n_i (\vec{\mu}_{pi} - \vec{\mu}_p)(\vec{\mu}_{pi} - \vec{\mu}_p)^T \quad (4.36)$$

de,

$$S_{PB} = V^T S_B V \quad (4.37)$$

ifadesiyle verilir.

Sınıflandırma işleminin optimum şekilde yapılması için, izdüşürülmüş sınıflar arası saçılma matrisi (S_{PB}) ile izdüşürülmüş sınıf içi saçılma matrisi (S_{PW}) skaler olmadıklarından, pay ve payda terimlerinin determinant olduğu

$$J(V) = \frac{\det(V^T S_B V)}{\det(V^T S_W V)} \quad (4.38)$$

biçiminde $J(V)$ ölçütü ile tanımlanır. Bu ölçüt maksimize edilerek en iyi ayıran hiperdüzlemler bulunur [67]. (4.38) ifadesinin maksimum kılınması, aşağıda verilen

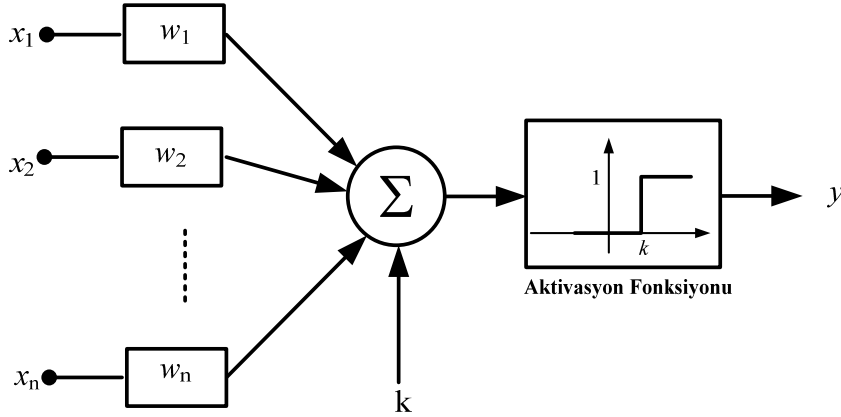
$$(S_B - \lambda_i S_W) \vec{v}_i = 0 \quad (4.39)$$

genelleştirilmiş özdeğer probleminin çözümüne indirgenir [68]. S_B matrisi için $\text{maksimum}(\text{rank}(S_B)) = c-1$ olmaktadır. İspatı Ek B’de verilmiştir. (4.39) ifadesi ile verilen özdeğer probleminin özdeğerleri ve karşı düşen özvektörleri $\vec{v}_i$, V izdüşüm matrisinin sütunlarını oluştururlar. Bu matrisler reel ve yarı kesin pozitif olduklarından özdeğerleri de pozitif reeldir ve maksimum değere sahip olanın özvektörü, verilerin sınıflandırılmasında en iyi ayrımı sağlamaktadır [64]. Birden fazla sayıda ve birbirine eşit maksimum özdeğer çıkması durumunda, bu maksimum özdeğerlerden elde edilen ve farklı olan özvektörlerden herhangi biri seçilip veri sınıflandırılmasında kullanılabilir (bu farklı özvektörlerin hepsi veriler arasında aynı derecede ayırım sağlamaktadır [68]). Maksimum özdeğere karşı düşen izdüşüm vektörü elde edildikten sonra sınıflandırma devresi ile kullanılması Bölüm 5.3’de ele alınmıştır.

4.2 Eğitici Perseptron Öğrenme Algoritması ile ÇAD ve ÇD Parametrelerinin Bulunması

Çift eşik hiperdüzlemlerinin oluşturulması için kullanılabilecek bir yöntem de yapay sinir ağlarında kullanılan eğitici perseptron öğrenme algoritmasıdır [28]. Bölüm 3’de ÇAD ve ÇD yapıları kullanılarak eğik ızgaralı verilerin, çift eşik hiperdüzlemi

ile nasıl sınıflandırılacağı incelenmiştir. Çift eşik hiperdüzlemi belirleyen katsayılar tek katlı perseptron öğrenme algoritmasından faydalanılarak da elde edilebilmektedir. Dolayısıyla perseptron öğrenme algoritması da Bölüm 3’de ele alınan verilerin sınıflandırılması için kullanılabilir. Şekil 4.4’de n girişli tek katlı perseptron yapısı verilmiştir.



Şekil 4.4 : n girişli tek katlı perseptron yapısı.

Şekil 4.4 ile verilen bu yapının tanım bağıntısı da şöyledir:

$$y = \begin{cases} 1 & \sum_{i=1}^n x_i w_i \geq k \quad \text{için} \\ 0 & \sum_{i=1}^n x_i w_i < k \quad \text{için} \end{cases} \quad (4.40)$$

Sınıflandırmada kullanılacak katsayılar (4.40) eşitliğindeki w ve k değerleridir. Bu w ve k değerleri ile hiperdüzlem denklemleri oluşturulur. Sınıflandırma için gerekli olan çift eşik hiperdüzlem denklemleri her farklı sınıf için aşağıdaki ifadeler kullanılarak elde edilir:

$$y_i = \begin{cases} 1 & \vec{x} \vec{v}_i^T - a \geq 0 \\ 0 & \vec{x} \vec{v}_i^T - a < 0 \end{cases} \quad (4.41)$$

$$y_i = \begin{cases} 1 & \vec{x} \vec{v}_i^T - b \leq 0 \\ 0 & \vec{x} \vec{v}_i^T - b > 0 \end{cases} \quad (4.42)$$

(4.41) ve (4.42) ifadelerinde kullanılan $\vec{v}$ vektörü, a ve b parametrelerinin değerleri perseptron öğrenme algoritmasına göre hesaplanır.

Öğrenme algoritması sonucu, hiperdüzlem denklemleri

$$\vec{x} \vec{v}_i^T - a = 0 \quad (4.43)$$

$$\vec{x} \vec{v}_i^T - b = 0 \quad (4.44)$$

biçiminde yazılabilir.

Bu yöntemde veri, sınıflara uygun sayıda çift eşik hiperdüzlemleri ile ayrıştırılarak sınıflandırılmaktadır.

Uygun sayıda çift eşik hiperdüzlem denkleminin bulunması aşağıdaki yöntemi adım adım uygulayarak yapılır:

c_i ($i=1,2,...,k$) veri sınıflarını gösterebilir,

1. Herhangi bir c_i sınıfını diğer tüm sınıflardan ayıran uygun bir adet çift eşik hiperdüzlemi olup olmadığı perseptron öğrenme algoritmasıyla kontrol edilir.
 - i. Eğer varsa, c_i sınıfı için ağırlık katsayıları kaydedilir. Madde 2'ye geçilir.
 - ii. Eğer bulunamıyorsa madde 3'e geçilir.
2. Bir sonraki sınıfa geçilir.
 - i. Eğer bir sonraki sınıf sonuncu k sınıfı ise sınıflandırma işlemi sonlandırılır.
 - ii. Değil ise madde 1 uygulanır.
3. Çift eşik hiperdüzlem sayısı bir artırılır ve c_i sınıfının diğer sınıflardan ayrılıp ayrılmadığı aynı algoritmayla kontrol edilir.
 - i. Ayrılabilir ise çift eşik hiperdüzlemlerinin ağırlık katsayıları kaydedilir. Madde 2'ye geçilir.
 - ii. Ayrılamıyor ise madde 3'e geçilir.

Perseptron yapısında kullanılan aktivasyon fonksiyonu keskin eşik fonksiyonu biçiminde olduğu için ağırlık katsayıları perseptron öğrenme algoritmasına göre güncellenerek elde edilir [14].

Perseptron öğrenme algoritması,

$$v_i(n+1) = v_i(n) - \eta(y_d - y_o)x_i \quad (4.45)$$

$$a_i(n+1) = a_i(n) - \eta(y_d - y_o) \quad (4.46)$$

$$b_i(n+1) = b_i(n) - \eta(y_d - y_o) \quad (4.47)$$

biçimindedir. Buradaki y_d elde edilmek istenilen sonuç, y_o ise elde edilen sonuç değeridir. η ise öğrenme katsayısı olarak isimlendirilir; ve 0 ile 1 arasında değer alır. Güncelleme işleminde $y_d = y_o$ olması durumunda ağırlık katsayısı değişmez. Öğrenme algoritması bütün ağırlık katsayıları değişmediği durumda durur [69].

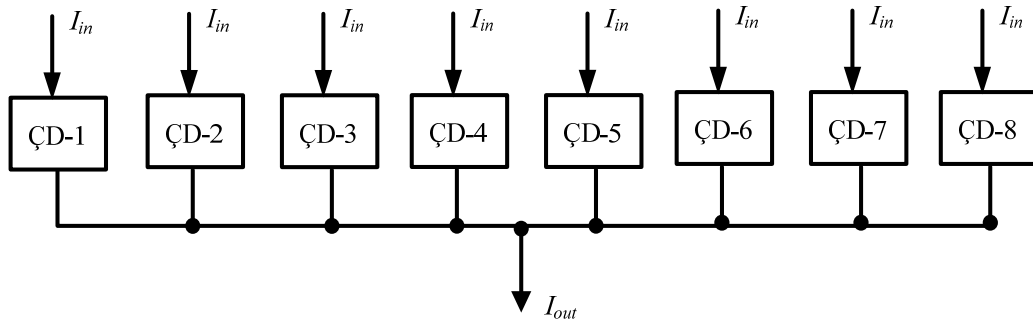
Eğitim kümesindeki verilerden c_i sınıfına ait veriler kullanılarak güncelleme işlemi bittikten sonra (4.41) ve (4.42) ifadeleri kullanılarak bulunmuş olan çift eşik hiperdüzlemi, $a_i \leq \vec{x} \vec{v}_i^T \leq b_i$ bölgesini göstermektedir ve c_i sınıfına ait veriler bu bölge içinde bulunmaktadır. Öğrenme algoritması ile bulunan $\vec{v}$ vektörü ÇAD parametreleri olan w 'lara (ağırlık katsayıları), a ve b ise çekirdek devrenin I_1, I_2 (kontrol akımları) değerlerine karşılık gelmektedir ve ÇD'nin I_H parametresinden yararlanılarak veri sınıfı belirlenmektedir. Şayet veri kümesindeki bir sınıf, iki veya daha çok sayıda çift eşik hiperdüzlemi gerektirirse, diğer bir deyişle iki veya daha çok sayıda bölgede aynı sınıfın olması durumunda, farklı bölgelere karşılık düşen aynı verilerin ÇD'lerinin I_H parametreleri aynı seçilerek veri sınıfının doğru sınıflanması sağlanır. Dolayısıyla uygun sayıda ÇAD ve ÇD bloğu kullanılarak sınıflandırma yapısı oluşturulur. Perseptron öğrenme algoritması kullanılarak üç ve dört boyutlu iki farklı veri kümesi Bölüm 5.3'de sınıflandırılmış ve sonuçları verilmiştir.

5. SINIFLANDIRICI DEVRE UYGULAMALARI

5.1 Kuantalayıcı

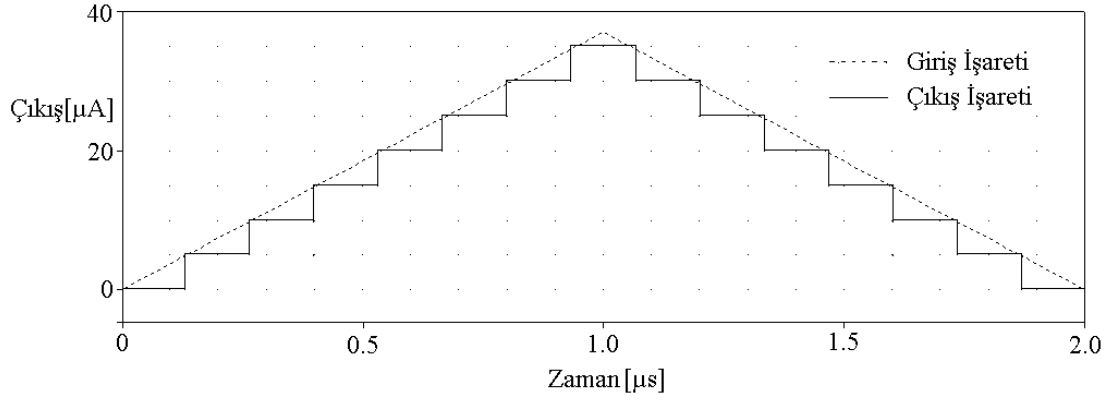
Gerçekleştirilmiş olan çekirdek devre yapıları kullanılarak kuantalayıcı devre oluşturulabilmektedir.

Sekiz seviyeli kuantalayıcı devre oluşturmak için sekiz adet çekirdek devre Şekil 5.1’de gösterildiği biçimde paralel olarak bağlanmıştır. Bu çekirdek devrelerin girişlerine aynı giriş işareti (kuantalanacak işaret) uygulanmıştır.



Şekil 5.1 : Sekiz seviyeli kuantalayıcı devre blok yapısı [55].

Kuantalayıcı yapı, aynı zamanda tek boyutlu 8 farklı veri türünü ayırt etmek için kullanılabilen bir sınıflandırıcı yapısıdır. Ancak yapının kuantalayıcı olarak çalışması için her bir çekirdek devrenin I_1 , I_2 ve I_H kontrol parametrelerinin Çizelge 5.1’de verildiği biçimde uygulanmış olması gerekir. Çizelgede verilen parametreler doğrultusunda kuantalayıcı yapının benzetimi için devrenin girişine üçgen bir dalga uygulanmıştır. Devrenin giriş-çıkış karakteristiği Şekil 5.2’de verilmiştir. Ayrıca kuantalama devresinde kuantalama aralıkları ve kuantalama seviyesi kontrol akımlarına bağlı olarak değiştirilebilir.



Şekil 5.2 : Sekiz seviyeli kuantalayıcı devresi $I_{in}-I_{out}$ karakteristiği.

Çizelge 5.1 : Kuantalayıcı Yapısında Kullanılan Çekirdek Devre Parametreleri.

ÇD	1	2	3	4	5	6	7	8
I_1 (µA)	0	5	10	15	20	25	30	35
I_2 (µA)	40	40	40	40	40	40	40	40
I_H (µA)	0.1	5	5	5	5	5	5	5

Gerçekleştirilen kuantalayıcı yapı ile çekirdek devrenin kontrol parametrelerinin dışarıdan değiştirilebiliyor ve ayrıca ÇD bloklarının paralel bağlanabiliyor olmasının sağladığı avantajlar, bu çalışmada ele alınan yaklaşımın üstünlüğünü göstermesi açısından da yararlı olmaktadır.

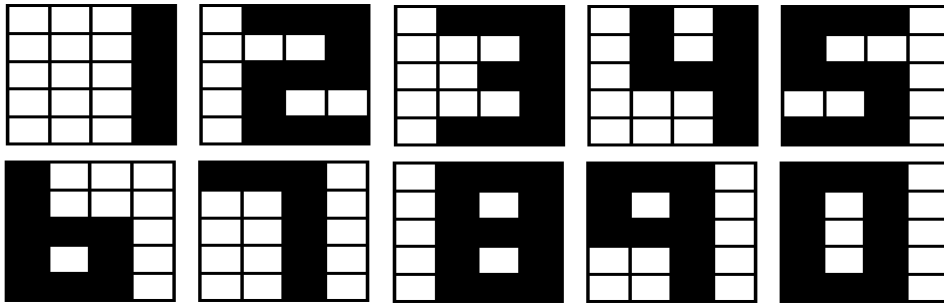
5.2 Karakter Tanıma

Gerçekleştirilen devrenin bir başka uygulaması olarak çok kullanılmakta olan “Karakter Tanıma” problemi şablon uydurma metodu kullanılarak ele alınmıştır. Karakter tanıma çok tipik bir sınıflandırma örneği olarak gösterilmektedir [1,70]. Buradaki sınıflandırma işlemi ele alınan verinin bir şablona uyup uymadığının saptanması şeklindedir. Burada verilen örnekte şablonlar sayı karakterleri olarak düşünülmüş ve Şekil 5.3’de verilmiştir. Karakter tanıma, örüntü tanımanın özel bir şeklidir. Şablon uydurmadan büyük bir resmin içindeki küçük resimleri tanımada yararlanır. Burada şablonun küçük resimler olduğu düşünülebilir. Hatta bu şablon herhangi bir karakter, resim ya da sayı olabilir. Her bir şablon Şekil 5.4’de gösterildiği gibi $4 \times 5 = 20$ hücreye bölünmüştür. Oluşabilecek bir satır hatasının düzeltilmesi için fazladan bir kolon kullanılmış ve karakterlerin yerleri özel olarak seçilmiştir.

Şekil 5.5’de gösterilen karakter tanıma sınıflandırıcısı, örnek şablonun, her hücrenin diğer örnek şablonlar ile karşılaştırılabileceği şekilde tasarlanmıştır [71]. Bu şekil aynı zamanda Dört Boyutlu Sınıflandırıcı Blokları (DBSB) içermektedir. DBSB yapısı, çıkışları aynı yere bağlanmış ve çıkış fonksiyonu $f(x_1, x_2, x_3, x_4)$ ile ifade edilen dört adet çekirdek devrenin paralel bağlanmasından oluşmaktadır. Her şablonun her satırının farklı bir sonuç vermesi için DBSB yapısının çıkışı,

$$f(x_1, x_2, x_3, x_4) = x_1 2^0 + x_2 2^1 + x_3 2^2 + x_4 2^3 \quad (5.1)$$

şeklinde seçilmiştir. Her DBSB’nin çıkışı, geçiş karakteristiği $g_i(y)$ fonksiyonu ile ifade edilen bir çekirdek devreye uygulanmıştır. Giriş örnek örüntüsünün satırları Şekil 5.5’de gösterildiği biçimde DBSB bloklarına uygulanır. Uygulanan giriş örüntüsüne göre, a_i, b_i, c_i, d_i ($i=1, \dots, 5$) değerleri sayısal olarak “1” veya “0” olabilir ve DBSB yapısının çıkışı (5.1) ifadesine göre değerler alır. Bu değerler Çizelge 5.2’de verilmektedir. Bu çizelgeden görüldüğü gibi çıkışlara ilişkin satırların biri, diğeri ile çakışmamaktadır. Böylece y_i ($i=1, \dots, 5$) çıkışlarını kontrol ederek giriş şablonunun ne olduğu söylenilebilir. Karakter tanıma işlemi bu aşamada gerçekleşmektedir. İstenen sadece karakter tanıma işleminin gerçekleştirilmesi ise, çıkışlar toplanarak tek bir çıkış ile elde edilebilir. Σy_i ile gösterilen bu toplam her bir giriş için farklı sonuç vermektedir.

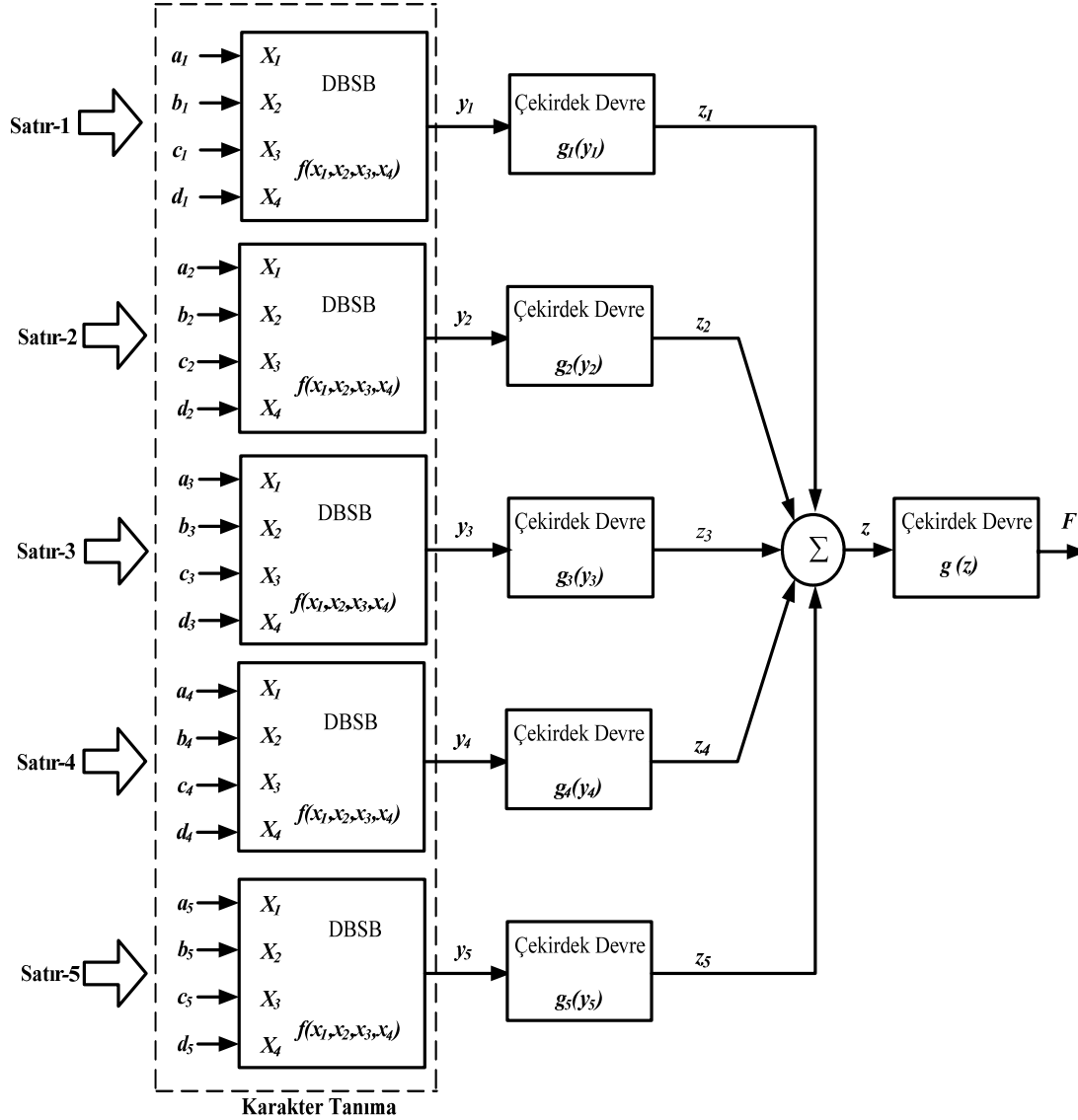


Şekil 5.3 : Örnek örüntüler.

a_1	b_1	c_1	d_1
a_2	b_2	c_2	d_2
a_3	b_3	c_3	d_3
a_4	b_4	c_4	d_4
a_5	b_5	c_5	d_5

Şekil 5.4 : Şablon hücre gösterimi.

Karakter tanıma sınıflandırıcısında, $g_i(y_i)$ fonksiyonu ile gösterilen bloğun kontrol akımları, seçilmek istenen şablonu gösterecek şekilde ayarlanır. Ayırt edilmek istenen örüntü girişe uygulanması durumunda, ya da $g_i(y_i)$ çekirdek devresinin girişi uygun bölgede ise, z_i çıkışı sayısal olarak “1” değerini aksi durumda ise “0” değerini verecektir. Sonuç olarak, bulunmak istenen doğru şablon girişe uygulandığında “z” çıkışı genlik olarak ”5” değerini, farklı bir şablon uygulandığında da “5” den daha küçük bir değeri verecektir.



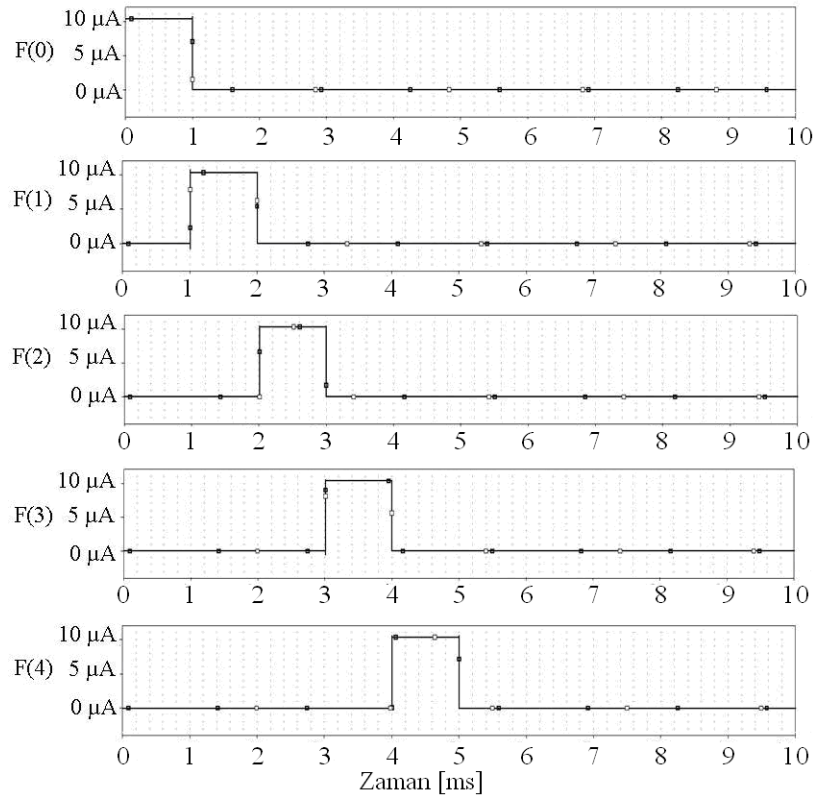
Şekil 5.5 : Karakter tanıma sınıflandırıcısı blok diyagramı.

Ayrıca Çizelge 5.2’de, herhangi iki farklı giriş için en fazla üç adet y_i ($i=1,2,...,5$) çıkışının birbiri ile çakıştığı açık bir şekilde görülmektedir. Bu durumdan hata düzeltilmesinde yararlanılabilir.

Çizelge 5.2 : Farklı örüntüler için y_i ($i=1,\dots,5$) çıkış değerleri.

Şablonlar	y_1	y_2	y_3	y_4	y_5	Σy_i
0	7	5	5	5	7	29
1	8	8	8	8	8	40
2	14	8	14	2	14	52
3	14	8	12	8	14	56
4	10	10	14	8	8	50
5	7	1	7	4	7	26
6	1	1	7	5	7	21
7	7	4	4	4	4	23
8	14	10	14	10	14	62
9	7	5	7	4	4	27

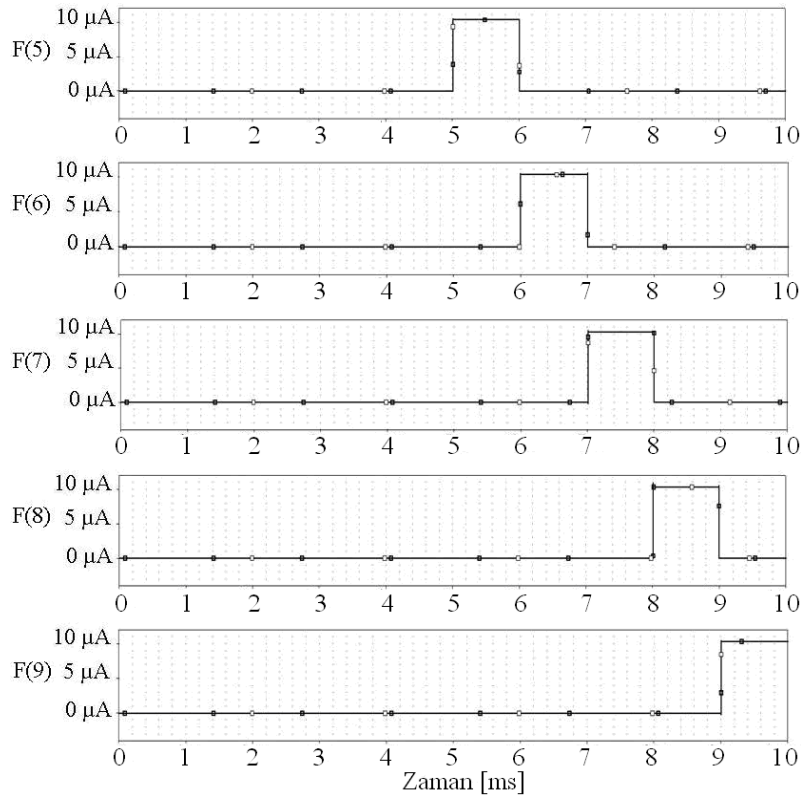
Sınıflandırıcı devresinin, şablon uydurma yöntemi ile karakter tanıma uygulaması için Şekil 5.3’de verilen şablonlar ile benzetimi yapılmıştır. Bunun için bir şablon seçilmiş, 0 ms ile 1ms arasında şablon “0”, 1 ms ile 2 ms arasında şablon “1” ve diğer şablonlar 10 ms olana kadar benzer şekilde uygulanmış; benzetim 10 şablon seçilerek tekrarlanmıştır. Benzetimde “1” genliği için 10 μA değeri seçilmiştir. Karakter tanıma uygulamasının benzetim sonuçları, ilk beş şablon için Şekil 5.6’da son beş şablon için ise Şekil 5.7’ de verilmiştir.



Şekil 5.6 : İlk beş şablon için karakter tanıma sınıflandırıcısı benzetim sonuçları.

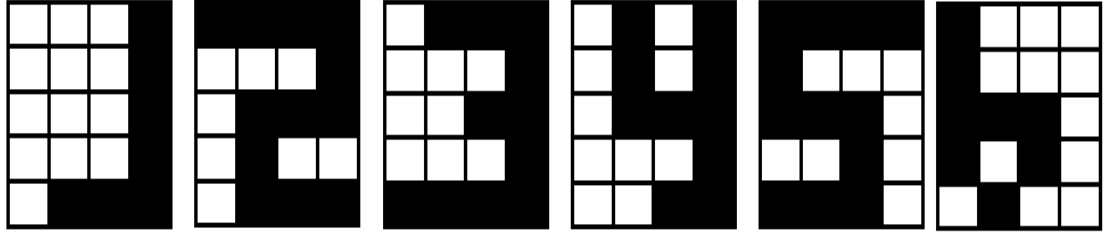
Bu şekillerden de görüldüğü gibi 0 ms ile 10 ms arasında sadece 1 ms’lik bir kısmın genliği 10 μ A olmakta ve devrenin istenildiği gibi çalıştığı görülmektedir.

Çizelge 5.2’deki sadece bir satırın, bir veya daha çok hücresinde hata oluştuğunda z_i çıkışı genlik olarak ”4” değerini alacaktır. Oluşan hatanın düzeltilmesi için sınıflandırıcı bloğunun sonuna $g(z)$ fonksiyonunu sağlayan bir adet çekirdek devre eklenebilir. Bu çekirdek devre oluşan girişine, ”4” veya ”5” genlikli bir işaret geldiğinde çıkışta genlik olarak ”1” verilmesini dolayısıyla da hatanın düzeltilmesini sağlamış olacaktır.

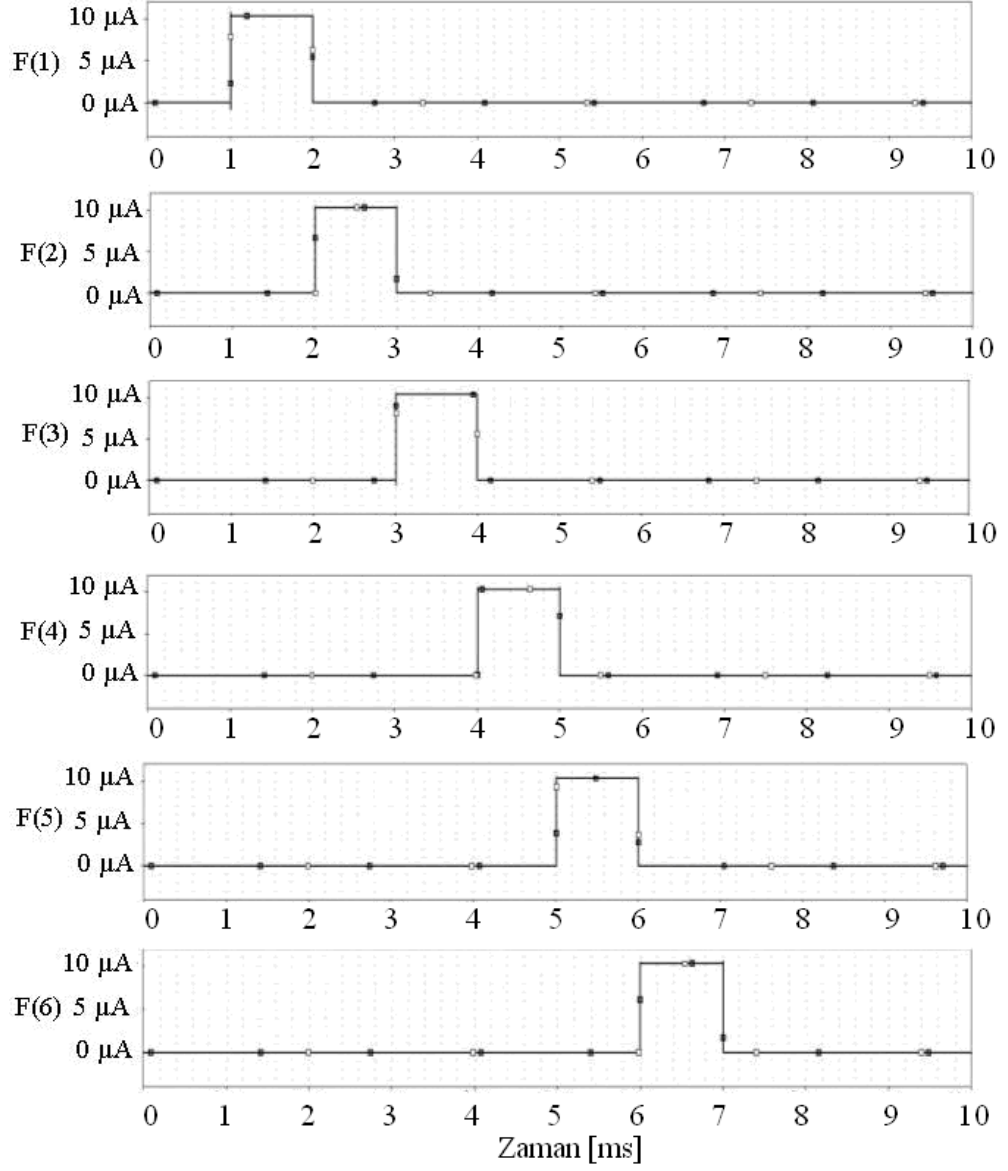


Şekil 5.7 : Son beş şablon için karakter tanıma sınıflandırıcısı benzetim sonuçları.

Karakter tanıma sınıflandırıcısının, aynı zamanda hata düzeltme işlemini de gerçekleştirdiğini göstermesi açısından, Şekil 5.8’de verilen hatalı şablonlar, 1 ms ile 2 ms arasında şablon ”1”, 2 ms ile 3 ms arasında şablon ”2” ve diğerleri, 7 ms olana kadar benzer şekilde bloğa uygulanmıştır. Benzetim sonucu Şekil 5.9’da gösterilmiştir. Sonuçtan da görüldüğü gibi test şablonlarının tek bir satırında, tek ya da daha çok hücrede, hata olması durumunda doğru sınıflandırma işlemi gerçekleşmiştir [71].



Şekil 5.8 : Hatalı test şablonları.



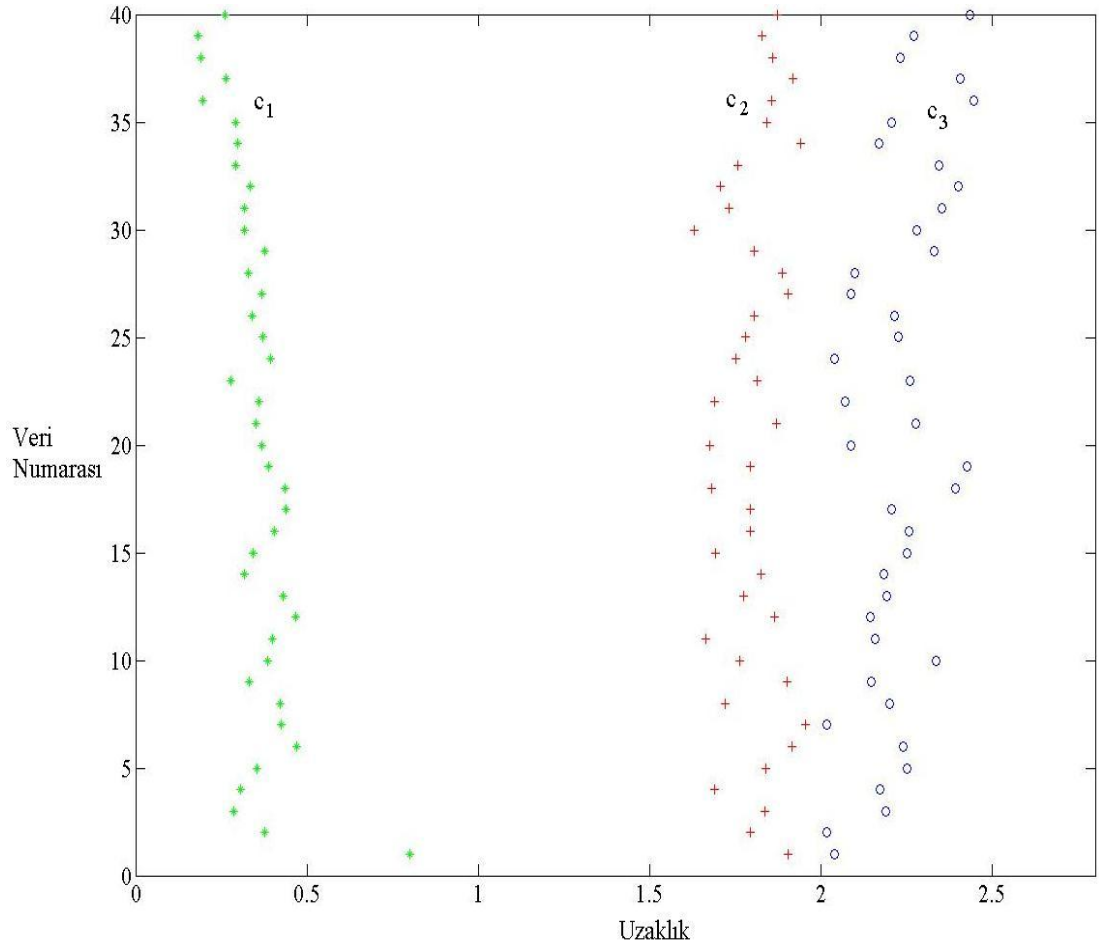
Şekil 5.9 : Karakter tanıma sınıflandırıcısı hata düzeltme benzetim sonucu.10

Bu örnekle, şablon uydurma yöntemi ile karakter tanıma uygulaması için gerçekleştirilmiş olan sınıflandırıcı devre yapılarının şablon tanıma ve benzeri uygulamalarda da kullanılabileceği gösterilmiştir.

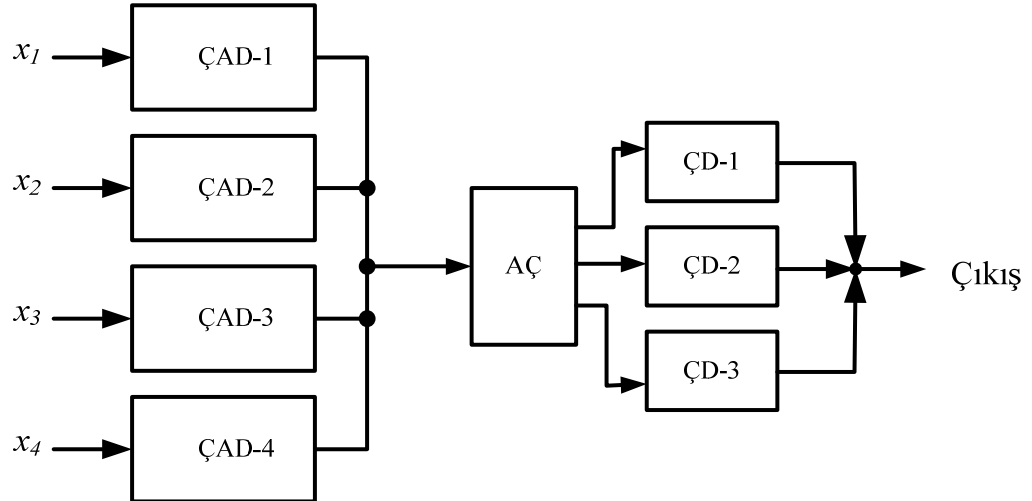
5.3 İris ve Haberman Verisinin Sınıflandırılması

5.3.1 İris verisinin Fisher tabanlı algoritma ile sınıflandırılması

Literatürde çok kullanılan veri kümelerinden biri iris veri kümesidir. Bu veri kümesi 1936 yılında Fisher tarafından oluşturulmuştur [72]. Sınıflandırma uygulamalarında kullanılan en temel veri sınıflarındandır. Veri kümesi dört adet özellik, diğer bir ifadeyle dört boyutlu veri (x_1, x_2, x_3, x_4) ve üç sınıftan (c_1, c_2, c_3) oluşmaktadır. Bu sınıflar Setosa, Versicolor ve Virginica isimli bitki tipleridir. Veri kümesinin sınıflandırılması bitkinin taç ve çanak yapraklarının uzunluk ve kalınlık değerlerine bakılarak gerçekleştirilir. İris veri kümesi 150 adet veri içermektedir. Bu veriler üç sınıfa eşit sayıda dağılmıştır. Veri kümesi detaylı olarak Ek C1’de verilmiştir. Sınıflandırma için kullanılacak veri kümesinin özellikleri santimetre birimi cinsinden verilmiş ve gerçekleştirilmiş olan sınıflandırıcı ile test edilmiştir. Bu testte sınıflandırıcının kontrol parametrelerinin, Bölüm 4.1’de geliştirilmiş olan Fisher tabanlı algoritma ile hesaplanması için, her sınıftan 40 veri ve toplamda 120 veri kullanılmıştır. Fisher tabanlı algoritma sonucunda veriler arasında en iyi ayrımı sağlayacak izdüşüm vektörü $\vec{v} = (0.57 \quad -0.80 \quad 0.10 \quad 0.14)$ olarak elde edilmiştir. Bu vektör (4.4) ifadesinde belirtildiği şekilde kullanılarak iris veri kümesinin orijine olan izdüşüm uzaklığı hesaplanarak verilerin orijine olan izdüşüm uzaklıkları Şekil 5.10’da gösterilmiştir. Bu şekil üstündeki, yıldız işaretleri c_1 , artı işaretleri c_2 ve küçük daireler de c_3 sınıfına ait verilerdir. Şekil 5.10’dan görüldüğü gibi yatay eksen, verilerin orijine olan izdüşüm uzaklıklarını ve dikey eksen ise veri numarasını göstermektedir. Dolayısıyla Şekil 5.10’daki verilerin uzaklıkları ile iris veri kümesi sınıflandırılabilir. Bu sınıflandırma işlemi için ÇAD ve ÇD yapılarından faydalanılmıştır. Şekil 5.11’de iris verisi sınıflandırıcısı blok diyagramı verilmiştir. Bu yapıdaki her bir ÇAD yapısının ağırlık katsayısı w , Fisher tabanlı algoritma ile elde edilen $\vec{v}$ vektörünün elemanlarına denk düşmektedir. ÇAD yapılarının çıkış değişkeni akım boyutunda olduğundan çıkış uçları birbirlerine bağlanarak toplama işlemi gerçekleştirilir. Bu toplam, veri kümesinin orijine olan izdüşüm uzaklığını gösterir. Dolayısıyla sınıflandırma işlemi için izdüşüm uzaklıkları (akım değerleri) üç çıkışlı Akım Çoğullayıcı (AÇ) devre ile çekirdek devre girişlerine uygulanmıştır. Şekil 5.10’da gösterildiği biçimde, her bir sınıfın ayrıldığı noktalar çekirdek devrenin kontrol akım değerlerine karşılık gelmektedir.



Şekil 5.10 : İzdüşürülmüş iris verilerinin orijine uzaklıkları.



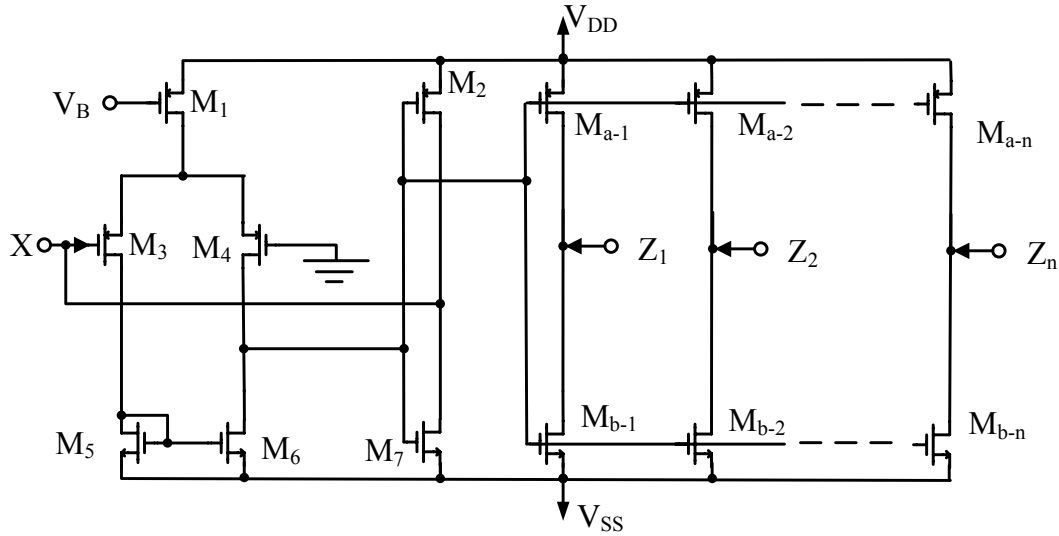
Şekil 5.11 : İris verisi sınıflandırıcı bloğu (Fisher tabanlı algoritma sonucu oluşturulmuş).

Şekil 5.11'deki yapının gerçekleştirilmesi için kullanılan akım çoğullayıcının n çıkışlı devre şeması Şekil 5.12'de ve tanım bağıntısı da (5.2) ifadesi ile verilmiştir. Akım çoğullayıcı sayesinde, ÇAD akımlarının toplamı tekrar çekirdek devrelere

uygulanarak, Şekil 5.10’da gösterilen bölgeler, çıkışın üç farklı değerden birini almasıyla seçilir.

$$\begin{bmatrix} V_x \\ I_{z-1} \\ I_{z-2} \\ I_{z-3} \\ \vdots \\ \vdots \\ \vdots \\ I_{z-n} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & \dots & 0 \\ 1 & 0 & 0 & 0 & \dots & 0 \\ 1 & 0 & 0 & 0 & \dots & 0 \\ 1 & 0 & 0 & 0 & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & 0 & 0 & 0 & \dots & 0 \end{bmatrix} \begin{bmatrix} I_x \\ V_{z-1} \\ V_{z-2} \\ V_{z-3} \\ \vdots \\ \vdots \\ \vdots \\ V_{z-n} \end{bmatrix} \quad (5.2)$$

Şekil 5.11’deki devrenin benzetiminde kullanılan n çıkışlı akım çoğullayıcı yapı için 0.35 μm AMS CMOS teknoloji parametreleri ve besleme gerilimleri V_{DD} ve V_{SS} ± 1.65 V olarak kullanılmıştır. Kutuplama akımının 18 μA olması için V_B gerilimi 0.8 V olarak alınmıştır. Akım çoğullayıcı devresinde kullanılan tranzistor boyutları da Çizelge 5.3’de verilmiştir.



Şekil 5.12 : Akım çoğullayıcı devre şeması [65].

Çizelge 5.3 : n çıkışlı akım çoğullayıcı devresi MOS tranzistorların boyutları.

MOSFET	W [μm]	L [μm]
M_1, M_3, M_4	21	1.05
M_5, M_6	6.3	1.05
$M_2, M_{a-1}, M_{a-2}, \dots, M_{a-n}$	42.7	1.05
$M_7, M_{b-1}, M_{b-2}, \dots, M_{b-n}$	11.2	1.05

Sınıflandırıcının başarımı için her sınıftaki test kümesinden geri kalan 10, toplamda 30 adet veri kullanılarak Şekil 5.11’de gösterilen yapının SPICE benzetimi yapılmıştır. Benzetim için kullanılan test verileri Çizelge 5.4’de gösterilen aralıklarla Şekil 5.11’deki yapıya uygulanmıştır.

Çizelge 5.4 : Şekil 5.11’deki iris verisi sınıflandırıcısı test kümesi.

Verilerin uygulanma süresi	x_1	x_2	x_3	x_4	Sınıf
0 ms - 1 ms	4.3	2.3	1.4	0.2	c_1
1 ms - 2 ms	5.7	2.7	3.9	1.1	c_2
2 ms - 3 ms	5.7	2.7	4.8	1.8	c_2
3 ms - 4 ms	4.9	2.2	6	2.5	c_3
4 ms - 5 ms	5.6	2.5	5.1	1.9	c_3
5 ms - 6 ms	4.6	3	1.7	0.4	c_1
6 ms - 7 ms	4.7	3.1	1.5	0.1	c_1
7 ms - 8 ms	6.1	2.9	3.8	1.1	c_2
8 ms - 9 ms	6.4	3	4	1.3	c_2
9 ms - 10 ms	4.9	2	4.7	1.4	c_2
10 ms - 11 ms	4.8	3.2	1.2	0.2	c_1
11 ms - 12 ms	5.4	3.8	1.6	0.6	c_1
12 ms - 13 ms	6.3	2.8	6.5	1.8	c_3
13 ms - 14 ms	6.7	3	6.4	2	c_3
14 ms - 15 ms	7.2	3.2	5.4	2.1	c_3
15 ms -16 ms	5.4	3.9	1.9	0.4	c_1
16 ms - 17 ms	5.3	3.7	1.5	0.2	c_1
17 ms - 18 ms	5.4	3.8	1.3	0.3	c_1
18 ms - 19 ms	6.3	3.0	4.4	1.3	c_2
19 ms - 20 ms	6.3	3.0	4.1	1.3	c_2
20 ms - 21 ms	6.9	3.1	5.6	2.4	c_3
21 ms - 22 ms	7.6	3.3	5.7	2.5	c_3
22 ms - 23 ms	7.7	3.4	5.2	2.3	c_3
23 ms - 24 ms	6.4	3.0	4.0	1.3	c_2
24 ms - 25 ms	6.4	3.0	4.4	1.2	c_2
25 ms - 26 ms	6.5	3.0	4.0	1.2	c_2
26 ms - 27 ms	7.7	3.4	5.0	1.9	c_3
27 ms - 28 ms	7.7	3.6	5.2	2.0	c_3
28 ms - 29 ms	5.4	3.8	1.6	0.6	c_1
29 ms - 30 ms	5.4	3.9	1.9	0.4	c_1

Çizelge 5.4’de verilmiş olan iris verisinin özellikleri olan x_i ($i=1,2,\dots,4$) değerleri cm boyutundadır. Ancak benzetim için bu değerler ÇAD sayesinde gerekli normalizasyon yapılarak akıma dönüştürülmüş ve çekirdek devreye uygulanmıştır. Çizelge 5.4’deki 0-1 ms arasındaki $x_1=4.3$ verisinin ÇAD-1 yapısında kullanılması şöyle özetlenebilir: Şekil 3.20’deki yapıda $V_y=0.43$ V ve $R_1=10$ k Ω alınması durumunda giriş büyüklüğü $V_y / R_1 = 43 \mu A$ olarak akıma çevrilir. Ayrıca Fisher tabanlı algoritma ile bulunan $\vec{v} = (0.57 \quad -0.80 \quad 0.10 \quad 0.14)$ vektörünün ilk

elemanı 0.57 olduğu için $R_2/R_3 = 57 \text{ k}\Omega / 100 \text{ k}\Omega$ alınarak (3.6) ifadesine göre çıkış akımı elde edilir. Bu çıkış değeri elde edilirken $x_1=4.3$ olan giriş verisi μA boyutunda gibi düşünülüp 10 kat büyütülerek $43 \mu\text{A}$ olarak $\vec{v}$ vektörünün elemanı ile çarpılmıştır. Benzer durum diğer giriş verileri içinde uygulandığından, orijine olan izdüşüm uzaklığı Şekil 5.10'da gösterilmiş olan uzaklığın 10 katı olarak elde edilmiştir. Bu durumda Şekil 5.10'da c_1 sınıfına ait verilerin orijine olan izdüşüm uzaklığı 0.1 ile 0.8 arasında olduğu halde verilerin ayırt edilebilmesi için Şekil 5.11'deki çekirdek devre-1'in kontrol akım değerleri $I_1=1 \mu\text{A}$, $I_2=8 \mu\text{A}$ olacak şekilde seçilmiştir. Şekil 5.11'de kullanılan diğer ÇAD ve ÇD parametreleri benzer şekilde bulunmuş ve sırası ile Çizelge 5.5'de ve Çizelge 5.6'da verilmiştir. Ayrıca Şekil 5.11'de kullanılan ÇAD çıkışları $\vec{v}$ vektörünün elemanlarının pozitif ya da negatif olmasına göre seçildiğinden, Şekil 3.20'den farklı olarak tek çıkışları bulunmaktadır. Diğer bir ifadeyle $\vec{v}$ vektörünün ilk elemanı pozitif, ikinci elemanı negatiftir, dolayısıyla Şekil 5.11'de kullanılan ÇAD-1 yapısı için pozitif çıkış ucu, ÇAD-2 yapısı için negatif çıkış ucu kullanılmıştır.

Çizelge 5.5 : İris sınıflandırıcısı ÇAD direnç değerleri.

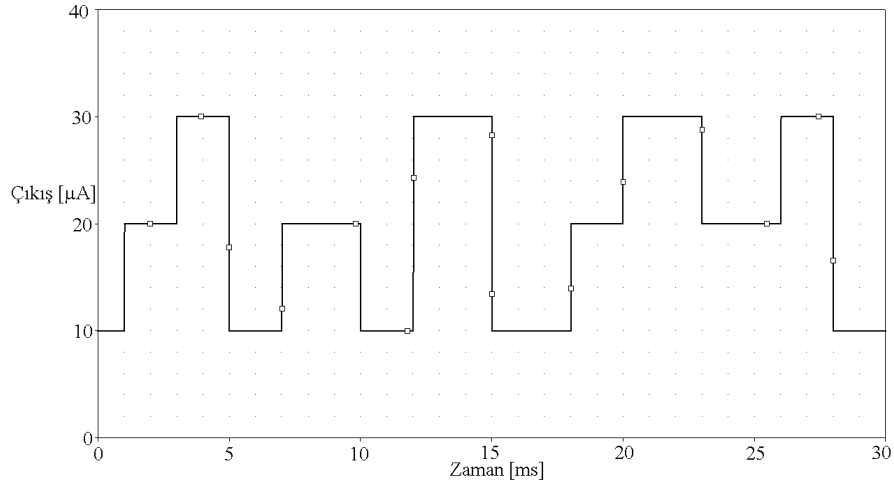
Dirençler	ÇAD-1	ÇAD-2	ÇAD-3	ÇAD-4
R_1	10 k Ω	10 k Ω	10 k Ω	10 k Ω
R_2	57 k Ω	80 k Ω	10 k Ω	14 k Ω
R_3	100 k Ω	100 k Ω	100 k Ω	100 k Ω

Şekil 5.11'deki yapının çıkış akımı çekirdek devrelerin I_H akımları ile kontrol edildiğinden her çekirdek devrenin I_H akımı farklı seçilerek veri kümesinin sınıflandırılması sağlanır. Ayrıca iris kümesindeki veriler sınıflandırıcıya uygulanmadan önce sınıflandırıcının uygun çalışma bölgesi göz önüne alınarak 10 kat büyütülmüştür. Böylece ÇAD yapısı ile sınıflandırıcı devrenin çalışma bölgesi dışındaki veri kümelerinin de ayırt edilebildiği gösterilmiştir.

Çizelge 5.6 : İris sınıflandırıcısı çekirdek devre kontrol akımları.

Kontrol akımları	ÇD-1	ÇD-2	ÇD-3
I_1	1 μA	16 μA	21 μA
I_2	8 μA	19 μA	27 μA
I_H	10 μA	20 μA	30 μA

Çizelge 5.4’de sınıflandırıcı sonuçlarının ne çıkması gerektiği gözükmemektedir. Şekil 5.11’deki yapının benzetim sonucu Şekil 5.13’de verilmiştir; buna göre benzetim sonucunda çıkış değerinin 10 μ A olması durumunda sonuç c_1 , 20 μ A olması durumunda sonuç c_2 , 30 μ A olması durumunda sonuç c_3 olmaktadır.



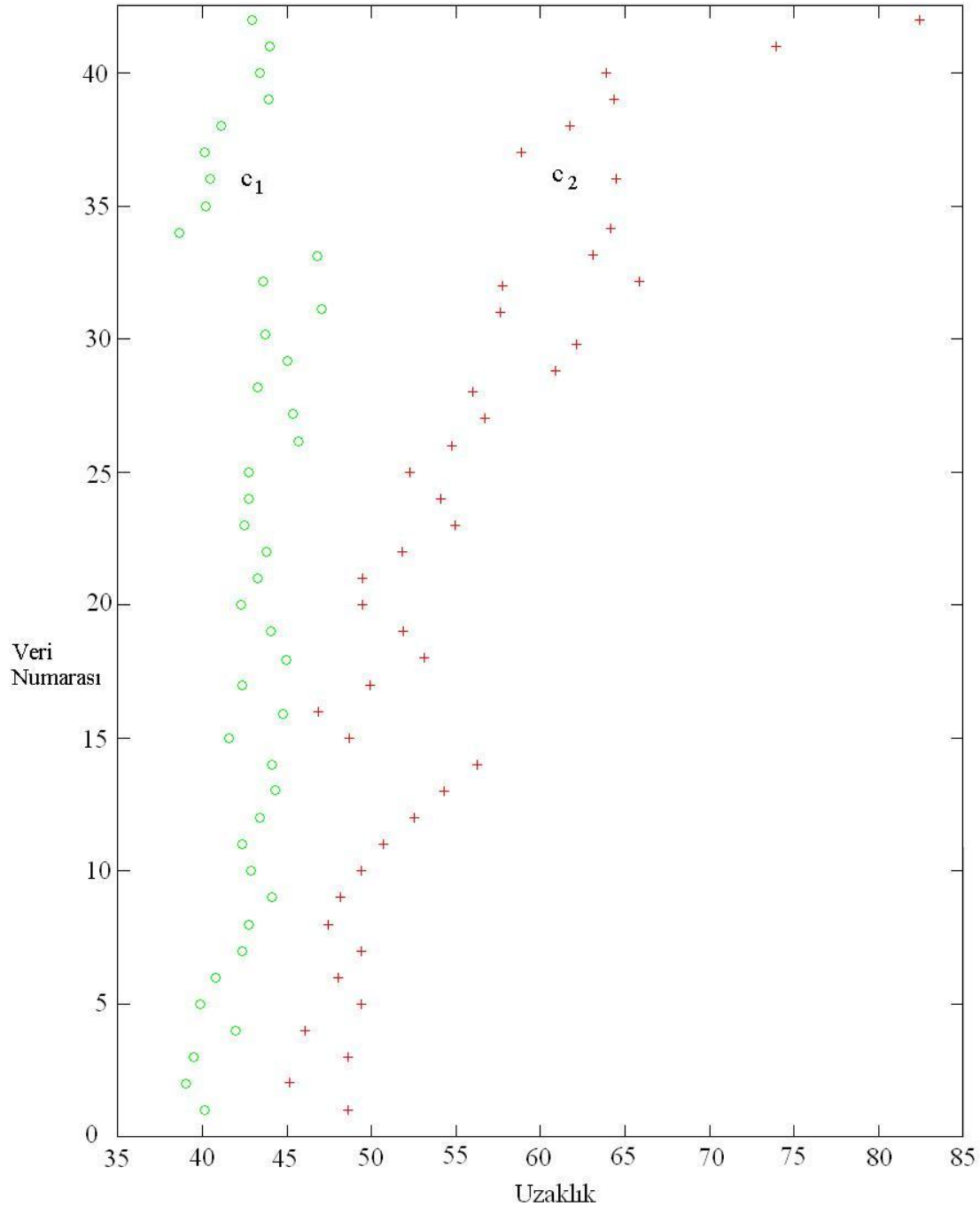
Şekil 5.13 : Şekil 5.11’deki İris verisi sınıflandırıcısı benzetim sonucu.

Şekil 5.13’de benzetim sonucundan görüldüğü gibi 30 adet iris test verisi doğru olarak sınıflandırılmıştır ve bu veri için Bölüm 4.1’de verilen Fisher tabanlı algoritma kullanılarak veri kümesinin sınıflandırılabilmesi sağlanmıştır.

5.3.2 Haberman verisinin Fisher tabanlı algoritma ile sınıflandırılması

Literatürde sınıflandırmada çok kullanılan diğer bir veri kümesinde Haberman veri kümesidir [73]. Bu veri kümesi üç boyutlu (x_1, x_2, x_3) olup iki sınıftan (c_1, c_2) oluşmaktadır. Bu küme Haberman tarafından 1958-1970 yılları arasında Şikago Üniversitesi Billing Hastanesinde göğüs kanseri teşhisi ile ameliyat olmuş hastalardan yararlanılarak oluşturulmuştur. Küme ameliyat sonrasında beş yıldan daha uzun yaşayan ya da beş yıl içinde ölen hastaları sınıflandırmaktadır. Sınıflandırma işlemi için hastanın yaşı, ameliyat tarihi ve pozitif netice veren göğüs kanserli hasta sayısı kullanılmıştır; veri kümesi toplam 306 adet veri içermekte ve bu verilerin 255 adeti c_1 sınıfına, 81 adeti ise c_2 sınıfına ait bulunmaktadır. Sınıflandırıcıların başarımlarının karşılaştırılmasında kolaylık sağlaması açısından iki sınıfa ait rastgele 104 veri alınmış ve sınıflara eşit dağıtılmıştır. Bu verilerden her sınıftan 42 ve toplamda 84 veri kullanılarak, Bölüm 4.1’de ele alınan Fisher tabanlı algoritma Haberman veri kümesine uygulanmıştır. Fisher tabanlı algoritma

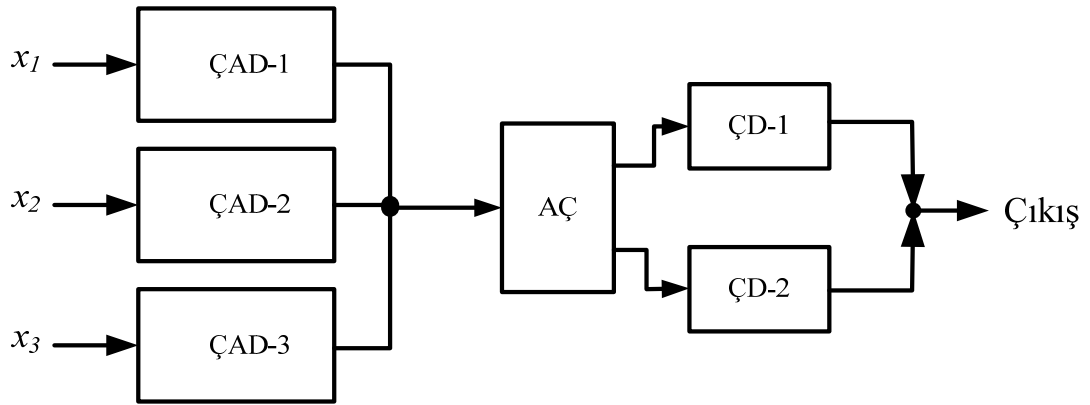
sonucunda izdüşüm vektörü $\vec{v} = (0.23 \ 0.50 \ 0.80)$ olarak elde edilmiştir. Böylece bu izdüşüm vektörü (4.4) ifadesinde gösterildiği şekilde kullanılarak Haberman veri kümesinin orijine olan izdüşüm uzaklıkları bulunmuş ve Şekil 5.14'deki şekilde çizdirilmiştir. Bu şekilde, küçük yuvarlaklar c_1 ve artı işaretleri c_2 sınıfına ait verilerdir.



Şekil 5.14 : İzdüşürülmüş haberman verilerinin orijine uzaklıkları.

Şekil 5.14'den görüldüğü gibi yatay eksen verilerin orijinden uzaklıklarını dikey eksen ise veri numarasını göstermektedir. Dolayısıyla iris veri kümesinde olduğu

gibi, aynı yöntemle Şekil 5.14'deki verilerin uzaklıkları ile de Haberman veri kümesi de sınıflandırılabilir. Bu sınıflandırma işlemi için ÇAD ve ÇD yapılarından faydalanılmıştır. Şekil 5.15'de Haberman verisi sınıflandırıcısı blok diyagramı verilmiştir. ÇAD çıkışlarındaki akımların toplamı, girişe uygulanan verinin orijine olan izdüşüm uzaklığına karşılık gelmektedir. Dolayısıyla sınıflandırma işlemi için, izdüşüm uzaklığı (akım değeri) iki çıkışlı akım çoğullayıcı devre ile iki çekirdek devre girişine uygulanmıştır. Bu sayede çekirdek devreler ile Şekil 5.14'de gösterilen bölgelerin seçilmesi sağlanmaktadır.



Şekil 5.15 : Haberman verisi sınıflandırıcı bloğu (Fisher tabanlı algoritma sonucu oluşturulmuştur).

Çizelge 5.7'de verilmiş olan Haberman verisinin özellikleri olan x_i ($i=1,2,3$) değerleri μA akım boyutunda düşünülmüştür. Benzetim için Çizelge 5.7'deki 0-1 ms arasında $x_1=34$ verisinin ÇAD-1 yapısında kullanılması şu şekilde özetlenebilir: Şekil 3.20'deki yapıda $V_y=34$ mV ve $R_1=10$ k Ω alınması durumunda giriş büyüklüğü akımı $V_y/R_1 = 34 \mu A$ olarak seçilmiştir. Ayrıca Fisher tabanlı algoritmanın uygulanmasıyla bulunan $\vec{v} = (0.23 \ 0.50 \ 0.80)$ vektörünün ilk elemanı 0.23 olduğu için, $R_2/R_3 = 23 \ k\Omega / 100 \ k\Omega$ alınarak (3.6) ifadesine göre çıkış akımı elde edilir. Diğer taraftan Şekil 5.14'de c_1 sınıfına ait verilerin orijine olan izdüşüm uzaklığı 38 ile 45 arasında olduğu için Şekil 5.15'deki çekirdek devre-1'in kontrol akımı değerleri $I_1=38 \ \mu A$, $I_2=45 \ \mu A$ olacak şekilde seçilmiştir. Benzer şekilde Şekil 5.15'de kullanılan ÇAD ve ÇD parametreleri bulunmuş ve sırası ile Çizelge 5.8'de ve Çizelge 5.9'da verilmiştir. Ayrıca her çekirdek devrenin I_H akımı farklı seçilmiştir. Bu sayede Şekil 5.15'deki yapının çıkış akımına bakılarak veri sınıfının bulunması sağlanmıştır.

Çizelge 5.7 : Şekil 5.15’deki Haberman verisi sınıflandırıcısı test kümesi.

Verilerin uygulanma süresi	x_1	x_2	x_3	Sınıf
0 ms - 1 ms	34	60	1	c_1
1 ms - 2 ms	61	68	1	c_2
2 ms - 3 ms	51	59	3	c_2
3 ms - 4 ms	37	59	6	c_1
4 ms - 5 ms	54	58	1	c_1
5 ms - 6 ms	61	62	5	c_2
6 ms - 7 ms	42	63	1	c_1
7 ms - 8 ms	53	61	1	c_1
8 ms - 9 ms	48	67	7	c_2
9 ms - 10 ms	65	66	15	c_2
10 ms - 11 ms	60	59	17	c_2
11 ms - 12 ms	42	59	2	c_1
12 ms - 13 ms	30	62	3	c_1
13 ms - 14 ms	65	62	22	c_2
14 ms - 15 ms	41	60	23	c_2
15 ms - 16 ms	46	58	3	c_1
16 ms - 17 ms	42	61	4	c_1
17 ms - 18 ms	72	67	3	c_1
18 ms - 19 ms	47	63	23	c_2
19 ms - 20 ms	43	58	52	c_2

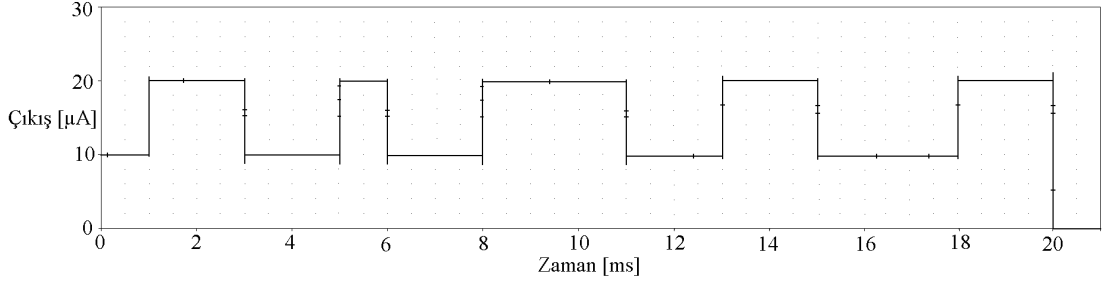
Çizelge 5.8 : Haberman sınıflandırıcısı ÇAD direnç değerleri.

Dirençler	ÇAD-1	ÇAD-2	ÇAD -3
R_1	10 k Ω	10 k Ω	10 k Ω
R_2	2.3 k Ω	5 k Ω	8 k Ω
R_3	10 k Ω	10 k Ω	10 k Ω

Çizelge 5.9 : Haberman sınıflandırıcısı çekirdek devre kontrol akımları.

Kontrol akımları	ÇD-1	ÇD-2
I_1	38 μ A	45 μ A
I_2	46 μ A	82 μ A
I_H	10 μ A	20 μ A

Sınıflandırıcının başarımını sınamak için her sınıftan geri kalan 10 adet, toplamda 20 adet veri kullanılarak Şekil 5.15’deki sınıflandırıcı yapının SPICE benzetimi yapılmıştır. Benzetim için kullanılan test verileri Çizelge 5.7’de gösterilen aralıklar ile Şekil 5.15’deki yapıya uygulanmıştır. Çizelge 5.7’de sınıflandırıcı sonuçlarının ne olması gerektiği gözükmemektedir. Şekil 5.16’daki benzetim sonucu, çıkış değeri 10 μ A ise verinin c_1 , 20 μ A ise verinin c_2 sınıfına ait olduğunu göstermektedir.



Şekil 5.16 : Şekil 5.15'deki haberman verisi sınıflandırıcısı benzetim sonucu.

Şekil 5.16'da benzetim sonucundan görüldüğü gibi 20 adet Haberman test verisi Fisher tabanlı algoritma ile doğru olarak sınıflandırılmıştır.

5.3.3 İris verisinin perseptron öğrenme algoritması ile sınıflandırılması

İris veri kümesi, Bölüm 4.2'de ele alınan ÇAD ve ÇD parametrelerinin perseptron öğrenme algoritması kullanılarak bulunmasından faydalanılarak sınıflandırılmıştır. Öğrenme algoritması yardımı ile ÇAD ve ÇD parametrelerinin hesaplanması için her sınıftan 40 veri, toplamda 120 veri kullanılmıştır. Veri kümesi dört boyutlu ve sınıf sayısı üç olduğu için, Bölüm 4.2'de incelenen öğrenme algoritmasına göre 3 adet çift eşik hiperdüzlemi elde edilmiş olup, bu çift eşik hiperdüzlemlerinden biri c_1 sınıfını diğer ikisi ise c_2 sınıfının bulunduğu bölgeleri göstermektedir. Üç adet çift eşik hiperdüzleminin belirlediği bölgelerin hiçbirinde bulunmayan veriler ise üçüncü sınıfa ait veri olarak sınıflandırılmıştır. Bu algoritma sonucu, her sınıf için gereken $\vec{v}$ vektörü (ÇAD parametreleri w_i 'ler) ve a, b sayıları (ÇD parametreleri, I_1 ve I_2) bulunmuştur. Böylece c_1 sınıfına ait veriler için $\vec{v}_{c_1} = (0.25 \ 0.96 \ -1.50 \ -0.63)$ vektörü, $a=1$ ve $b=3.5$ olarak elde edilmiştir. Benzer şekilde c_2 sınıfına ait veriler için bulunan ilk $\vec{v}_{c_2-1} = (0.3 \ 1.0 \ -0.2 \ 3.7)$ vektörü, $a_1=7.8$ ve $b_1=9.3$ ve ikinci $\vec{v}_{c_2-2} = (0.4 \ 2.0 \ -0.1 \ 3.5)$ vektörü, $a_2=14$ ve $b_2=15$ olarak bulunmuştur. Elde edilen bu sonuçlar için oluşturulmuş olan iris sınıflandırıcısı bloğu Şekil 5.17'de verilmiştir.

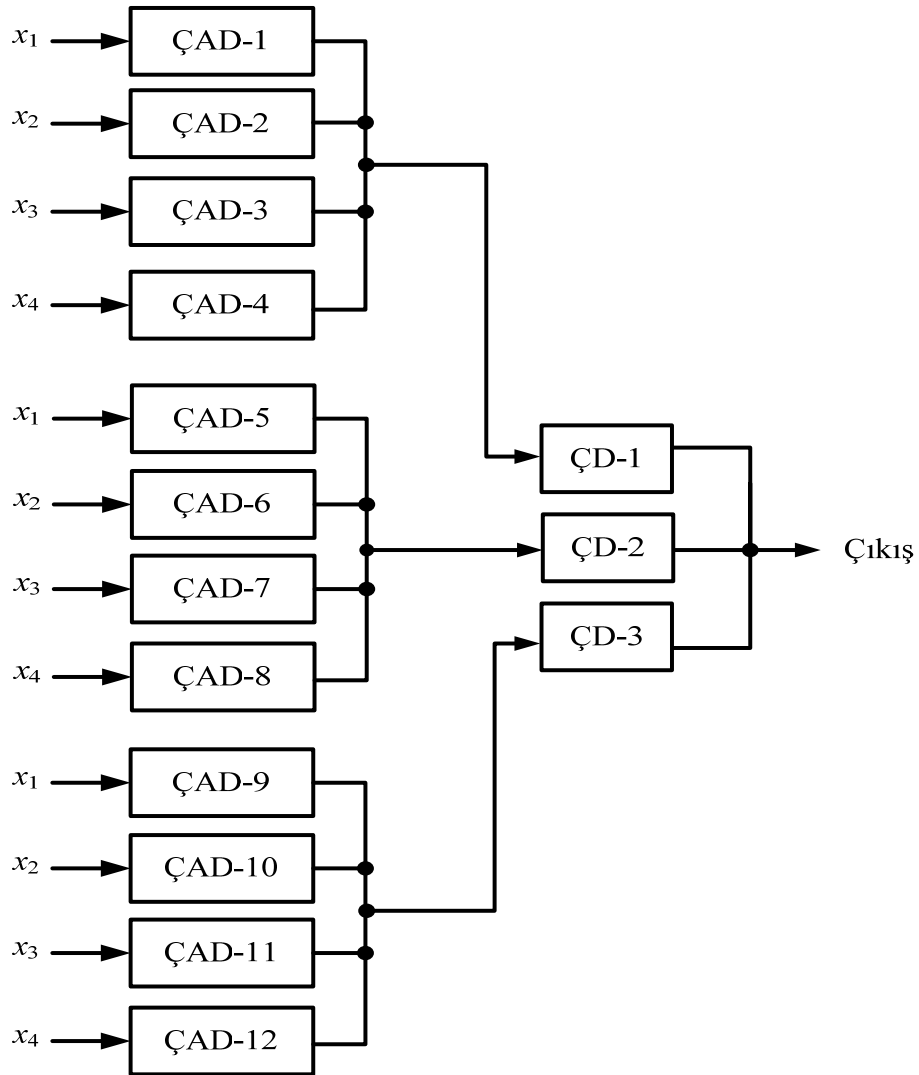
Sınıflandırıcının başarımını sınamak için her sınıftan geri kalan 10, toplamda 30 adet veri kullanılarak Şekil 5.17'de blok diyagramı verilen devrenin SPICE benzetimi yapılmıştır. Benzetim için kullanılan test verileri Çizelge 5.10'da gösterilen aralıklar ile Şekil 5.17'deki devreye uygulanmıştır.

Çizelge 5.10 : Şekil 5.17’deki iris verisi sınıflandırıcısı test kümesi.

Verilerin Uygulanma Süresi	x_1	x_2	x_3	x_4	Sınıf
0 ms - 1 ms	4.3	2.3	1.4	0.2	c_1
1 ms - 2 ms	5.7	2.7	3.9	1.1	c_2
2 ms - 3 ms	5.7	2.7	4.8	1.8	c_2
3 ms - 4 ms	4.9	2.2	6	2.5	c_3
4 ms - 5 ms	5.6	2.5	5.1	1.9	c_3
5 ms - 6 ms	4.6	3	1.7	0.4	c_1
6 ms - 7 ms	4.7	3.1	1.5	0.1	c_1
7 ms - 8 ms	6.1	2.9	3.8	1.1	c_2
8 ms - 9 ms	6.4	3	4	1.3	c_2
9 ms - 10 ms	4.9	2	4.7	1.4	c_2
10 ms - 11 ms	4.8	3.2	1.2	0.2	c_1
11 ms - 12 ms	5.4	3.8	1.6	0.6	c_1
12 ms - 13 ms	6.3	2.8	6.5	1.8	c_3
13 ms - 14 ms	6.7	3	6.4	2	c_3
14 ms - 15 ms	7.2	3.2	5.4	2.1	c_3
15 ms -16 ms	5.4	3.9	1.9	0.4	c_1
16 ms - 17 ms	5.3	3.7	1.5	0.2	c_1
17 ms - 18 ms	5.4	3.8	1.3	0.3	c_1
18 ms - 19 ms	6.3	3.0	4.4	1.3	c_2
19 ms - 20 ms	6.3	3.0	4.1	1.3	c_2
20 ms - 21 ms	6.9	3.1	5.6	2.4	c_3
21 ms - 22 ms	7.6	3.3	5.7	2.5	c_3
22 ms - 23 ms	7.7	3.4	5.2	2.3	c_3
23 ms - 24 ms	6.4	3.0	4.0	1.3	c_2
24 ms - 25 ms	6.4	3.0	4.4	1.2	c_2
25 ms - 26 ms	6.5	3.0	4.0	1.2	c_2
26 ms - 27 ms	7.7	3.4	5.0	1.9	c_3
27 ms - 28 ms	7.7	3.6	5.2	2.0	c_3
28 ms - 29 ms	5.4	3.8	1.6	0.6	c_1
29 ms - 30 ms	5.4	3.9	1.9	0.4	c_1

Çizelge 5.10’da verilmiş olan iris verisinin özellikleri olan x_i ($i=1-4$) değerleri cm boyutundadır. Ancak benzetimde bu değerler ÇAD sayesinde gerekli normalizasyon yapılarak akıma dönüştürülmüş ve çekirdek devreye uygulanmıştır. Benzetim yapılırken, Çizelge 5.10’daki 0-1 ms arasında $x_1=4.3$ verisinin, ÇAD-1 yapısında kullanılması şu şekilde özetlenebilir: Şekil 3.20’deki devrede $V_y=0.43$ V ve $R_1=10$ k Ω alınarak giriş büyüklüğü $V_y/R_1 = 43$ μ A olacak şekilde akım boyutuna dönüştürülür. Ayrıca öğrenme algoritması ile c_1 sınıfı için bulunan $\vec{v}_{c1} = (0.25 \ 0.96 \ -1.50 \ -0.63)$ vektörünün ilk elemanı 0.25 olduğu için, $R_2/R_3 = 25$ k Ω / 100 k Ω alınarak (3.6) ifadesine göre çıkış akımı elde edilir. Bu çıkış değeri elde edilirken, $x_1=4.3$ olan giriş verisi μ A boyutunda gibi düşünülüp 10 kat büyütülmüş ve 43 μ A olarak $\vec{v}_{c1}$ vektörünün ilk elemanı ile çarpılmıştır. Bu durumda c_1 sınıfı için bulunmuş olan $a=1$ ve $b=3.5$ değerleri (çekirdek devre I_1, I_2

parametresi) 10 kat büyük olacaktır. Diğer bir ifade ile Şekil 5.17'deki çekirdek devre-1'in kontrol akım değerleri $I_1=10 \mu A$, $I_2=35 \mu A$ olarak seçilmiştir. Benzer durum göz önünde alınarak Şekil 5.17'de kullanılan ÇAD ve ÇD parametreleri, c_1 ve c_2 sınıfı için bulunarak sırası ile Çizelge 5.11'de ve Çizelge 5.12'de verilmiştir. Ayrıca Şekil 5.17'deki ÇAD yapılarının çıkışları $\vec{v}_{ci}$ vektörünün elemanlarının pozitif ya da negatif olmasına göre seçilmiştir. Bu nedenle Şekil 3.20'den farklı olarak tek çıkış bulunmaktadır. Diğer bir ifadeyle $\vec{v}_{c1} = (0.25 \ 0.96 \ -1.50 \ -0.63)$ vektörünün ilk ve ikinci elemanı pozitif, üçüncü ve dördüncü elemanı negatiftir. Dolayısıyla Şekil 5.17'de kullanılan ÇAD-1 ve ÇAD-2 yapısı için pozitif çıkış ucu, ÇAD-3 ve ÇAD-4 yapısı için negatif çıkış ucu kullanılmıştır.



Şekil 5.17 : İris verisi sınıflandırıcı bloğu (öğrenme algoritması sonucu oluşturulmuş).

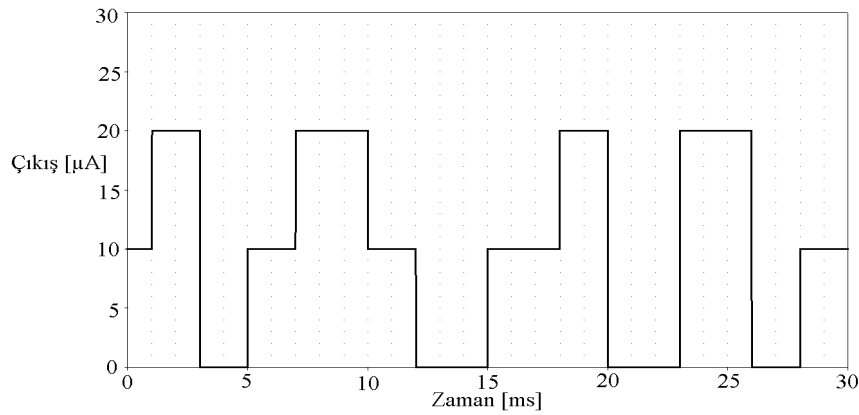
Çizelge 5.11 : Şekil 5.17’de c_1 sınıfı için ÇAD direnç değerleri ve ÇD akımları.

ÇAD-1	ÇAD -2	ÇAD -3	ÇAD -4	ÇD-1
$R_1=10\text{ k}\Omega$	$R_1=10\text{ k}\Omega$	$R_1=10\text{ k}\Omega$	$R_1=10\text{ k}\Omega$	$I_1=10\text{ }\mu\text{A}$
$R_2=2.5\text{ k}\Omega$	$R_2=9.6\text{ k}\Omega$	$R_2=15\text{ k}\Omega$	$R_2=6.3\text{ k}\Omega$	$I_2=35\text{ }\mu\text{A}$
$R_3=10\text{ k}\Omega$	$R_3=10\text{ k}\Omega$	$R_3=10\text{ k}\Omega$	$R_3=10\text{ k}\Omega$	$I_H=10\text{ }\mu\text{A}$

Çizelge 5.12 : Şekil 5.17’de c_2 sınıfı için ÇAD direnç değerleri ve ÇD akımları.

ÇAD-5	ÇAD-6	ÇAD-7	ÇAD-8	ÇD-2	ÇAD-9	ÇAD-10	ÇAD-11	ÇAD-12	ÇD-3
$R_1=10\text{ k}\Omega$	$R_1=10\text{ k}\Omega$	$R_1=10\text{ k}\Omega$	$R_1=10\text{ k}\Omega$	$I_1=78\text{ }\mu\text{A}$	$R_1=10\text{ k}\Omega$	$R_1=10\text{ k}\Omega$	$R_1=10\text{ k}\Omega$	$R_1=10\text{ k}\Omega$	$I_1=140\text{ }\mu\text{A}$
$R_2=3\text{ k}\Omega$	$R_2=10\text{ k}\Omega$	$R_2=2\text{ k}\Omega$	$R_2=37\text{ k}\Omega$	$I_2=93\text{ }\mu\text{A}$	$R_2=4\text{ k}\Omega$	$R_2=20\text{ k}\Omega$	$R_2=1\text{ k}\Omega$	$R_2=35\text{ k}\Omega$	$I_2=150\text{ }\mu\text{A}$
$R_3=10\text{ k}\Omega$	$R_3=10\text{ k}\Omega$	$R_3=10\text{ k}\Omega$	$R_3=10\text{ k}\Omega$	$I_H=20\text{ }\mu\text{A}$	$R_3=10\text{ k}\Omega$	$R_3=10\text{ k}\Omega$	$R_3=10\text{ k}\Omega$	$R_3=10\text{ k}\Omega$	$I_H=20\text{ }\mu\text{A}$

Şekil 5.17’deki yapının çıkış akımı çekirdek devrelerin I_H akımları ile kontrol edilmektedir. Dolayısıyla her çekirdek devrenin I_H akımı farklı seçilerek veri kümesinin sınıflandırılması sağlanmıştır. Benzetim için kullanılan test verileri, Çizelge 5.10’da gösterilen aralıklarla Şekil 5.17’deki yapıya uygulanmıştır. Çizelge 5.10’da sınıflandırıcı sonuçlarının ne çıkması gerektiği gözükmemektedir; benzetim sonucu Şekil 5.18’de verilmiştir. Bu şekildeki benzetim sonucunda çıkış değerinin 10 μA olması durumunda sonuç c_1 , 20 μA olması durumunda ise c_2 , 0 μA olması durumunda da c_3 olarak değerlendirilmiştir.



Şekil 5.18 : Şekil 5.17’deki iris verisi sınıflandırıcısı benzetim sonucu.

Şekil 5.18’deki benzetim sonucundan görüldüğü 30 adet test verisi doğru olarak sınıflandırılmıştır.

5.3.4 Haberman verisinin perseptron öğrenme algoritması ile sınıflandırılması

Haberman veri kümesi, Bölüm 4.2’de ele alınan ÇAD ve ÇD parametrelerinin perseptron öğrenme algoritması kullanılarak bulunmasından faydalanılarak sınıflandırılmıştır. Öğrenme algoritması yardımı ile ÇAD ve ÇD parametrelerinin

hesaplanması için her sınıftan 42, toplamda 84 adet veri kullanılmıştır. Veri kümesi üç boyutlu ve sınıf sayısı iki olduğu için, Bölüm 4.2’de incelenen öğrenme algoritmasına göre 2 adet çift eşik hiperdüzlemi elde edilmiştir. Bu çift eşik hiperdüzlemlerinin belirlediği bölge içinde kalan veri kümesi c_1 sınıfına ait verileri, kalmayanlar ise c_2 sınıfı veri kümesi olarak değerlendirilerek sınıflandırılmıştır. Öğrenme algoritması sonucu $\vec{v}$ vektörleri (ÇAD parametreleri, w_i ’ler) ve a , b sayıları (ÇD parametreleri I_1 ve I_2) bulunmuştur. Bu durumda c_1 sınıfına ait veriler için ilk $\vec{v}_1 = (-1.4 \ 2.2 \ -16)$ vektörü, $a_1=11$ ve $b_1=69$ ve ikinci olarak $\vec{v}_2 = (-1.3 \ 2.2 \ -10)$ vektörü, $a_2=20$ ve $b_2=24$ değerleri elde edilmiştir. Bu değerler için oluşturulmuş Haberman sınıflandırıcısı bloğu Şekil 5.19’da verilmiştir.

Şekil 5.19 : Haberman verisi sınıflandırıcı bloğu (perseptron öğrenme algoritması sonucu oluşturulmuş).

Çizelge 5.13 : Şekil 5.19'daki haberman verisi sınıflandırıcısı test kümesi.

Verilerin Uygulanma Süresi	x_1	x_2	x_3	Sınıf
0 ms - 1 ms	34	60	1	c_1
1 ms - 2 ms	61	68	1	c_2
2 ms - 3 ms	51	59	3	c_2
3 ms - 4 ms	37	59	6	c_1
4 ms - 5 ms	54	58	1	c_1
5 ms - 6 ms	61	62	5	c_2
6 ms - 7 ms	42	63	1	c_1
7 ms - 8 ms	53	61	1	c_1
8 ms - 9 ms	48	67	7	c_2
9 ms - 10 ms	65	66	15	c_2
10 ms - 11 ms	60	59	17	c_2
11 ms - 12 ms	42	59	2	c_1
12 ms - 13 ms	30	62	3	c_1
13 ms - 14 ms	65	62	22	c_2
14 ms - 15 ms	41	60	23	c_2
15 ms - 16 ms	46	58	3	c_1
16 ms - 17 ms	42	61	4	c_1
17 ms - 18 ms	72	67	3	c_1
18 ms - 19 ms	47	63	23	c_2
19 ms - 20 ms	43	58	52	c_2

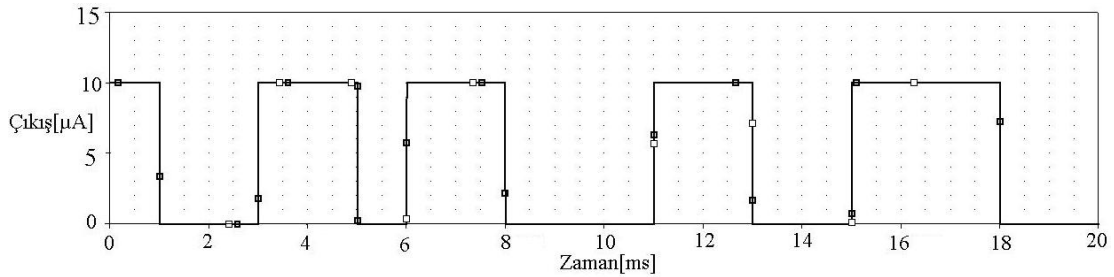
Çizelge 5.13'de verilmiş olan Haberman verisinin özellikleri olan x_i ($i=1,2,3$) değerleri μA boyutunda olarak düşünülmüştür. Benzetim için Çizelge 5.13'deki 0-1 ms arasındaki $x_1=34$ verisinin, ÇAD-1 yapısında kullanılması şu şekilde özetlenir: Şekil 3.20'deki yapıda $V_y=34$ mV ve $R_1=10$ k Ω alınması durumunda giriş büyüklüğü $V_y/R_1 = 34$ μA olarak akım boyutuna dönüştürülür. Ayrıca öğrenme algoritması ile c_1 sınıfı için bulunan $\vec{v}_1 = (-1.4 \quad 2.2 \quad -16)$ vektörünün ilk elemanı -1.4 olduğu için $R_2/R_3 = 14$ k Ω / 10 k Ω alınarak (3.7) ifadesine göre çıkış akımı elde edilir. Bu durumda c_1 sınıfı için bulunmuş olan $a=11$ ve $b=69$ değerleri (çekirdek devre I_1, I_2 parametresi) kullanılarak Şekil 5.19'daki çekirdek devre-1'in kontrol akım değerleri $I_1=11$ μA , $I_2=69$ μA olarak alınmıştır. Ayrıca Şekil 5.19'da kullanılan ÇAD yapılarının çıkışları $\vec{v}$ vektörünün elemanlarının pozitif ya da negatif olmasına göre seçilmiştir. Diğer bir ifadeyle $\vec{v}_1 = (-1.4 \quad 2.2 \quad -16)$ vektörünün ikinci elemanı pozitif, ilk ve üçüncü elemanı negatiftir. Dolayısıyla Şekil 5.19'da kullanılan ÇAD-2 bloğu için pozitif çıkış ucu, ÇAD-1 ve ÇAD-3 bloğu için ise negatif çıkış ucu kullanılmıştır. Şekil 5.19'daki yapının çıkış akımı çekirdek devrelerinin I_H akımı ile

kontrol edilmektedir. Dolayısıyla çekirdek devrelerinin I_H akımı, uygulanan veri tipinin c_1 sınıfı olup olmadığını belirlemektedir. Şekil 5.19’da kullanılan ÇAD ve ÇD devrelerinin parametreleri c_1 sınıfı için Çizelge 5.14’de verilmiştir.

Çizelge 5.14 : Şekil 5.19’da c_1 sınıfı için ÇAD direnç değerleri ve ÇD akımları.

ÇAD-1	ÇAD-2	ÇAD-3	ÇD-1	ÇAD-4	ÇAD-5	ÇAD-6	ÇD-2
$R_1=10\text{ k}\Omega$	$R_1=10\text{ k}\Omega$	$R_1=10\text{ k}\Omega$	$I_1=11\text{ }\mu\text{A}$	$R_1=10\text{ k}\Omega$	$R_1=10\text{ k}\Omega$	$R_1=10\text{ k}\Omega$	$I_1=20\text{ }\mu\text{A}$
$R_2=14\text{ k}\Omega$	$R_2=22\text{ k}\Omega$	$R_2=160\text{ k}\Omega$	$I_2=69\text{ }\mu\text{A}$	$R_2=13\text{ k}\Omega$	$R_2=22\text{ k}\Omega$	$R_2=100\text{ k}\Omega$	$I_2=24\text{ }\mu\text{A}$
$R_3=10\text{ k}\Omega$	$R_3=10\text{ k}\Omega$	$R_3=10\text{ k}\Omega$	$I_H=10\text{ }\mu\text{A}$	$R_3=10\text{ k}\Omega$	$R_3=10\text{ k}\Omega$	$R_3=10\text{ k}\Omega$	$I_H=10\text{ }\mu\text{A}$

Benzetim için kullanılan test verileri Çizelge 5.13’de gösterilen aralıklar ile Şekil 5.19’daki sınıflandırıcıya uygulanmıştır. Çizelge 5.13’de sonuçların ne çıkması gerektiği gözükmemektedir. Şekil 5.19’daki yapının benzetim sonucu da Şekil 5.20’de verilmiştir. Buna göre çıkış değerinin $10\text{ }\mu\text{A}$ olması durumunda sonuç c_1 , $0\text{ }\mu\text{A}$ olması durumundaysa sonuç c_2 olarak değerlendirilmiştir.



Şekil 5.20 : Şekil 5.19’daki Haberman verisi sınıflandırıcısı benzetim sonucu.

Şekil 5.20’deki benzetim sonucundan görüldüğü 20 adet Haberman test verisi doğru olarak sınıflandırılmıştır. İris ve Haberman verileri önerilen iki yöntem ile sınıflandırılmıştır. Sınıflandırıcı başarımları Çizelge 5.15’de verilmiştir. Önerilen algoritmalar ile sınıflandırıcı parametreleri Pentium 3 GHz işlemcili ve 1 GB belleği olan bir bilgisayar yardımı ile gerçekleştirilmiştir.

Çizelge 5.15 : Sınıflandırıcı başarımları karşılaştırması.

Algoritma	Sınıflandırıcı hata başarımları [%]		Algoritmanın sınıflandırıcı parametrelerini bulma süresi [sn]	
	İris	Haberman	İris	Haberman
Fisher tabanlı	100	100	0.4	0.3
Perseptron öğrenme	100	100	220	190

Çizelge 5.15’den görüldüğü gibi her iki yöntem ile de veri sınıflandırılabilir. Sınıflandırıcı algoritmasının kontrol parametrelerini bulması konusunda, Fisher tabanlı algoritmanın daha iyi olduğu gözlenmektedir. Ancak iki yöntem içinde kullanılan ÇD sayısının aynı olduğu görülmüştür.

Dolayısıyla gerçekleştirilmiş olan sınıflandırıcı devreleri ile literatürdeki veri kümelerinin başarılı bir şekilde sınıflandırıldığı, ayrıca sınıflandırma işlemi dışındaki diğer çeşitli uygulamalar için de kullanılabilecekleri ve yeni olanaklar ortaya çıkardıkları gösterilmiştir.

6. SONUÇLAR VE ÖNERİLER

Bu çalışmada, kuantalama, görüntü işleme, istatistik, tıbbi tanı, otomatik hedef belirleme, yapay zekâ, YSA, analog-sayısal dönüştürücüler gibi uygulama alanı bulan sınıflandırıcı devreler ve yeni olanakları ele alınmıştır. Literatürde rastlanan devrelerinin gerek güç tüketimi, gerek dışarıdan ayarlanabilme yeteneğine sahip olma gibi bazı sınıflandırma konularında istenildiği biçimde gereksinimleri karşılayamadığı saptanmıştır. Çalışmada, bu eksikliklerin bir kısmını giderecek ve mevcut literatüre katkı sağlayacak yeni devreler geliştirilmiş, bu devreler bazı gerçek uygulama yapıları ve benzetimler ile test edilip özgün sonuçlara ulaşılmıştır.

Bu çalışma kapsamında yapılanları şöyle sıralayabiliriz:

Gerek dışarıdan ayarlanabilme yeteneği, gerekse uygulama alanlarının esnek bir şekilde kullanılmasını sağlamak amacıyla önce temel bir sınıflandırıcı devre bloğuna işlevsel olarak karar verilmiş ve bu işlevi gerçekleyen iki adet CMOS devre önerilmiştir. Temel bir yapı niteliğinde olan bu sınıflandırıcı işlevsel bloğa çekirdek devre ismi verilmiştir. Bu çekirdek devrenin önemi temel bir yapı taşı gibi çeşitli topolojiler de kullanılabilir olmasıdır. Ayrıca, bu çekirdek devre literatürdeki mevcut yapılara göre çok daha esnek ve daha az güç tüketecek şekilde tasarlanmıştır [54]. Çekirdek devre blokları, akım modlu tasarlanmış olmalarından dolayı, bloklar uygun şekilde paralel ya da ard arda bağlanabilmektedir. Dolayısıyla tek boyutlu sınıflandırıcı yapılarından çok boyutlu sınıflandırıcı yapıların kolayca oluşturulmasına olanak tanınmıştır. Bu çok boyutlu sınıflandırıcı devreler ile dikedörtgen ızgara veri kümelerinin sınıflandırılacağı da gösterilmiştir ki bu tür veriler en yoğun kullanılanların başında gelmektedir. Özellikle, bazı veri kümeleri için çekirdek devrenin esnek yapıda tasarlanmasının getirdiği kullanım kolaylıkları da avantajları arasındadır. Ayrıca bahsedilen temel çekirdek yapısının, sınıflandırma bölgelerinin incelendiği ve literatürdeki mevcut sınıflandırma yapılarıyla karşılaştırılarak oldukça daha az güç harcadığı gösterilmiştir [58].

Diğer taraftan geliştirilen çok boyutlu sınıflandırıcı sadece dikdörtgen ızgara dağılımına sahip verilere uygulanabildiğinden, uygulanma alanlarını genişletebilmek amacıyla çekirdek devrelerin önüne girişin lineer kombinasyonunu alan yeni bir kat eklenmiş ve çarpan devre denilmiştir. Söz konusu ön kat, çok yoğun kullanılan bazı veri dağılımlarının (örneğin CMOS tranzistorların standart bölgeleri) sınıflandırılmasını da sağlamıştır [63]. Çarpan devrenin kontrol parametrelerini de kapsayacak şekilde sınıflandırma işlemi için gerekli parametrelerinin bulunmasını sağlayan algoritmalar da geliştirilmiş, üstünlükleri ve sakıncaları incelenmiştir.

Gerçekleştirilmiş olan sınıflandırıcı yapıların başarımlarına, gerçek dünyadaki uygulama alanlarına ve sağladığı yeni olanaklara örnek olarak kuantalayıcı devresi incelenmiştir [55]. Bunun yanı sıra, şablon uydurma yönteminin kullanıldığı ve bünyesinde oluşacak hataların düzeltilmesine de olanak tanıyan karakter tanıma uygulaması da gerçekleştirilmiştir [71].

Ayrıca sınıflandırıcı devrelerin, iki farklı öğrenme algoritması kullanılarak iki farklı veri kümesini sınıflandıran devrelerin parametreleri bulunmuş; benzetimlerle başarımları gösterilmiştir. Sonuçlar devrelerin ve algoritmaların etkinliğini ve başarımlarını göstermesi için karşılaştırmalı olarak verilmiştir.

Sonuç olarak, bu çalışmada gerçekleştirilmiş olan özgün sınıflandırıcı devrelerinin literatüre ve uygulamalara yeni olanaklar tanıdığı gösterilmiştir.

Bu tezin devamı niteliğinde dört farklı alanda yapılabilecek çalışmalardan ilki, daha kısa sürede eğitimin sağlanabileceği yeni öğrenme algoritmalarını geliştirmektir. İkincisi ise donanımsal olarak gerçekleştirilen devrelerde güç tüketimi ve kullanım kolaylığı açısından gelişme sağlanmasıdır. Üçüncü bir geliştirme, verileri ayıran daha genel bölgelerin sınıflandırılmasına olanak tanıyan bir ön kat ya da çekirdek hücrenin tasarımının yapılmasıdır. Dördüncüsü ve belki de en önemlisi, yazılımsal olarak geliştirilen algoritmaların donanımın içine dahil edilmesi, başka bir deyişle öğrenme sürecinin de donanımın içinde yapılmasıdır.

KAYNAKLAR

- [1] **Liu, B., Chen, C., and Tsao, J.**, 2000: A Modular Current-Mode Classifier Circuit for Template Matching Application, *IEEE Trans. On Circuit and Systems-II, Analog and Digital Sig. Process.*, Vol. **47**, pp. 145-151.
- [2] **Tzanakou, E. M.**, 2000: *Supervised and unsupervised Pattern Recognition*, CRC Press, Inc., pp. 7-75, Boca Raton, FL, USA.
- [3] **Cover, T. M., and Hart, P. E.**, 1967: Nearest neighbor pattern classification, *IEEE Trans. on Inform. Theory*, Vol. **13**, pp. 21-27.
- [4] **Bishop, C. M.**, 1996: *Neural Networks for Pattern Recognition*, Oxford University Press, Inc., pp. 116-191, Newyork, USA.
- [5] **Beiu, V.**, 1996: *Neural Networks and Their Application*, John Wiley & Sons Ltd., Baffins Lane, Chichester, pp. 1-98, England.
- [6] **Duda, R. O., Stork, D. G., and Hart, P. E.**, 2000: *Pattern Classification and Scene Analysis*, John Wiley & Sons Ltd., Inc., 2nd ed., pp. 1-57, England.
- [7] **Jain, A. K., Duin, R. P. W., and Mao, J.**, 2000: Statistical Pattern Recognition: A Review, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. **22**, pp. 4-37.
- [8] **Chenyz, Y., Hungyz, Y., and Fuhz, C.**, 2001: Fast Algorithm for Nearest Neighbor Search Based on a Lower Bound Tree, *Proceedings of the 8th International Conference on Computer Vision*, Vancouver, Canada, July 7-14, pp. 446-453.
- [9] **Abdel-Aty-Zohdy, H. S., and Al-Nsour, M.**, 1999: Reinforcement learning neural network circuits for electronic nose, *IEEE International Symposium on Circuits and Systems*, Florida, USA, May 30- July 2, pp. 379- 382.
- [10] **Ripley, B. D.**, 1996: *Pattern Recognition and Neural Networks*, Cambridge University Press, pp. 143-177, Cambridge, UK.
- [11] **Yamasaki, T., Yamamoto, K., and Shibata, T.**, 2001: Analog pattern classifier with flexible matching circuitry based on principal-axis-projection vector representation, *Proceedings of the 27th European Solid-State Circuits Conference*, Villach, Austria, September 18-20, pp. 197-200.

- [12] **Yamasaki, T., and Shibata, T.**, 2003: Analog soft-pattern-matching classifier using floating-gate MOS technology, *IEEE Trans. on Neural Networks*, Vol. **14**, pp. 1257-1265.
- [13] **Reyes, C., and Adjouadi, M.**, 1995: A Clustering Technique for Random Data Classification, *IEEE International Conference on Systems, Man and Cybernetics*, Vancouver, Canada, December 22-25, pp. 316-321.
- [14] **Zhao, Y., Deng, B., and Wang, Z.**, 2002: Analysis and Study of Perceptron to Solve XOR Problem, *Proceeding of the 2th International Workshop on Autonomous Decentralized System*, China, November 6-7, pp. 168-173.
- [15] **McCulloch, W. S., and Pitts, W.**, 1943: A logical calculus of the immanent in nervous activity, *Bulletion of Mathematical Biophysics*, Vol. **5**, pp. 115-133.
- [16] **Galushkin, A.I., and Sudarikov, V.A.**, 1992: Adaptive neural network algorithm for solving linear algebra problems, *IEEE Symposium on Neuroinformatics and Neurocomputers*, Rostov, Russia, October 7-10, pp. 128 – 138.
- [17] **Moody, J., and Darken C.**, 1989: Fast Learning in Networks of Locally-Tuned Processing Units, *Neural Computation*, Vol. **1**, pp. 281- 294.
- [18] **Xiaoqin, Z., Yajuan, H., and Yeung, D. S.**, 2003: Determining the relevance of input features for multilayer perceptrons, *IEEE International Conference on Systems Man ans Cybernetics*, October 5-8, pp. 874-879.
- [19] **Sun, P., and Makro K.**, 1997: A Neural Network Classifier For Conflicting Information Environments, *International Conference on Neural Networks*, Stocholm, Sweden, June 9-12, pp. 1617-1622.
- [20] **Hung, Y. C., and Liu B. D.**, 1999: A CMOS Analog Vector Quantizer For Pattern Recognition, *The First IEEE Asia Pacific Conference on ASICs*, Korea, August 23-25, pp. 112-115.
- [21] **Zhou, Z., Chen, S., and Chen, Z.**, 2000: FANNC: A Fast Adaptive Neural Network Classifier, *Knowledge and Informaton Systems*, Vol. **2**, pp. 115-129.
- [22] **Camboni, F., and Valle, M.**, 2001: A Mixed Mode Perceptron Cell for VLSI Neural Networks, *International Conference on Electronics, Circuits and Systems*, Malta, September 2-5, pp. 377-380.
- [23] **Jing, Y.**, 2006: An Improved Cascade SVM Training Algorithm with Crossed Feedbacks, *First International Multi-Symposiums Computer and Computational Sciences*, Istanbul, Turkey, June 20-24, pp. 735-738.

- [24] **Oyang, Y. J., Hwang S. C., Ou, Y.Y., Chen, C. Y., and Chen, Z. W.,** 2005: Data Classification with Radial Basis Function Networks based on a Novel Kernel Density Estimation Algorithm, *IEEE Transactions on Neural Networks*, Vol. **16**, pp. 225-236.
- [25] **Lubkin, J., and Cauwenberghs G.,** 1998: A Micropower Learning Vector Quantizer for Parallel Analog-to-Digital Data Compression, *Proceedings of the International Symposium on Circuits and Systems*, Vol **3**, May 31-June 3, pp. 58-61.
- [26] **Kachare, M., Carvajal, J. R. A., and Lopez, M. A. J.,** 2005: New Low-Voltage Fully Programmable CMOS Triangular/Trapezoidal Function Generator Circuits, *IEEE Trans. On Circuit and Systems-I*, Vol. **52**, pp. 2033-2040.
- [27] **Bilgili, E., Gökner, İ. C., and Uçan, O. N.,** 2005: Cellular Neural Networks with Trapezoidal Activation Function, *Int. Journal of Circuit Theory and Applications*, Vol. **33**, pp. 393-417.
- [28] **Aksın, D. Y., Aras, S., and Gökner, İ. C.,** 2000: CMOS Realization of User Programmable, Single-Level, Double-Threshold Generalized Perceptron, *Proceedings of Turkish Artificial Intelligence and Neural Networks Conference*, İzmir, Turkey, July 21-23, pp. 117-125.
- [29] **Bilgili, E., Gökner, C., Uçan, O. N., and Albora, M.,** 2006: Stability of CNN with Trapezoidal Activation Function, *Complex Computing-Networks: Brain-like and Wave-oriented Electrodynamical Algorithms Springer Proceedings in Physics*, Vol. **104**, pp. 267-274.
- [30] **Gray, R. M., and Neuhoﬀ, D. L.,** 1998: Quantization, *IEEE Trans. on Information Theory*, Vol. **44**, pp. 2325-2383.
- [31] **Leong, P. H. W., and Jabri, M. A.,** 1995: A Low Power VLSI Arrhythmia Classifier, *IEEE Transaction on Neural Networks*, Vol. **6**, pp. 1435-1445.
- [32] **Aybay, I., Çetinkaya, S., and Halıcı, U.,** 1996: Classification of Neural Networks Hardware, *Neural Networks World*, Vol. **6**, pp. 11-29.
- [33] **Erkmen, B.,** 2007: Genel Amaçlı Bir Yapay Sinir Ağı'nın Karma Bir Donanım ile Gerçeklenmesi, *PhD Thesis*, Yıldız Teknik Üniversitesi, İstanbul.
- [34] **Avcı, M.,** 2005: Hibrit Bir Donanımla Yapay Sinir Ağı Gerçekleme AY Tümevresi, *PhD Thesis*, Yıldız Teknik Üniversitesi, İstanbul.
- [35] **Cilingiroglu, U.,** 1993: A charge based neural Hamming Classifier, *IEEE Journal of Solid-State Circuits*, Vol. **28**, pp. 59-67.
- [36] **Aksın, D. Y., and Aras, S.,** 2005: A compact Distance Cell for Analog Classifiers, *Proceedings of the IEEE International Symposium on Circuits and Systems*, Kobe, Japan, May 23-26, pp. 3627-3630.

- [37] **Donckers, N., Dualibe, C., and Verleysen, M.,** 1999: Design of Complementary Low-Power CMOS Architectures for Looser-take all and Winner-take all, *Proceedings of the Seventh International Conference on Microelectronics for Neural, Fuzzy and Bio-Inspired Systems*, Granada, Spain, April 7-9, pp. 360-365.
- [38] **Cilingiroglu, U., and Aksin, D. Y.,** 1998: A 4 transistor Euclidean Distance Cell for Analog Classifiers, *Proceedings of the IEEE International Symposium on Circuits and Systems*, Monterey, USA, May 31- June 3, pp. 84-87.
- [39] **Grant, D., Taylor, J., and Houselander, P.,** 1994: A High-Speed Integrated Hamming Neural Classifier, *Proceedings of the IEEE International Symposium on Circuits and Systems*, London, England, May 30 –June 2, pp. 479-482.
- [40] **Kumar, N., Pouliquen, P. O., and Andreou, A. G.,** 1993: Device Mismatch Limitations on Performance of a Hamming Distance Classifier, *The IEEE International Workshop on Defect and Fault Tolerance in VLSI System*, Venice, Italy, October 27-29, pp. 327-334.
- [41] **Onat, B. M., McNeil, J. A., and Çilingiroğlu, U.,** 1997: Implementation of a charge based Neural Euclidian Classifier for a 3 bit flash Analog to digital converter, *IEEE Transactions on Instrumentation and Measurement*, Vol. **46**, pp. 672-677.
- [42] **Draghici, S., and Miller, D. A.,** 1999: A VLSI Neural Network Classifier Based on Integer-Valued Weights, *International Joint Conference on Neural Networks*, Washington, USA, July 10-16, pp. 2419-2424.
- [43] **Almeida, A. P., and Franca, J. E.,** 1993: A mixed-mode architecture for implementation of analog neural networks with digital programmability, *Proceedings of the International Joint Conference on Neural Networks*, Nagoya, Japan, October 25-29, pp. 887-890.
- [44] **Genov, R., and Cauwenberghs, G.,** 2003: Kerneltron: Support Vector Machine in Silicon, *IEEE Transactions on Neural Networks*, Vol. **14**, pp. 1426-1434.
- [45] **Chakbarrtty, S., and Cauwenberghs, G.,** 2007: Sub-Microwatt Analog VLSI Trainable Pattern Classifier, *IEEE Journal of Solid-State Circuits*, Vol. **42**, pp. 1169-1178.
- [46] **Peng, S. Y., Minch, B. A., and Hasler, P.,** 2008: Analog VLSI Implementation of Support Vector Machine Learning and Classification, *Proceedings of the IEEE International Symposium on Circuits and System*, Washington, USA, May 18-21, pp. 860-863.
- [47] **Peng, S. Y., Hasler, P. E., and Anderson, D.,** 2007: An Analog Programmable Multi-Diemensional Radial Basis Function Based Classifier, *International Conference on Very Large Scale Integration*, Atlanta, USA, October 15-17, pp. 13-18.

- [48] **Narayan, R., Honbo, D., Memik, G., Choudhary, A., and Zambreno, J.,** 2007: An FPGA Implementation of Decision Tree Classification, *Design, Automation & Test in Europe Conference & Exhibition*, Yokohama, Japan, April 16-20, pp. 1-6.
- [49] **Reaz, M. B. I., Islam, S. Z., Ali, M. A. M., and Sulaiman, M. S.,** 2002: FPGA Realization of Backpropagation for Stock Market Prediction, *Proceedings of the 9th International Conference on Neural Information*, Sofia, Bulgaria, Vol. 2, November 18-22, pp. 960-964.
- [50] **Wang, Y., and Salam, F. M. A.,** 1991: Experiments Using CMOS Neural Networks Chips as Pattern/Character Recognizers, *Proceedings of the IEEE International Symposium on Circuits and Systems*, Singapore, June 11-14, pp. 1196-1199.
- [51] **Bridges, S., Holleman, J., and Figueroa, M.,** 2006: A Random Projection Imager for Visual Pattern Classification in Analog VLSI, *Proceedings of the 32nd European Solid-State Circuits Conference*, Montreux, Switzerland, September 18-22, pp. 428-431.
- [52] **Bermak, A., and Martinez, D.,** 2003: A Compact 3-D VLSI Classifier Using Bagging Threshold Network Ensembles, *IEEE Transaction on Neural Networks*, Vol. 14, pp. 1097-1109.
- [53] **Djahanshahi, H., Jullien, G. A., Miller, W.C., and Ahmadi, M.,** 1996: Neural-Based Smart CMOS Sensors for On-Line Pattern Classification Application, *Proceedings of the IEEE International Symposium on Circuits and Systems*, Atlanta, USA, May 12-15, pp. 384-387.
- [54] **Yıldız, M., Minaei, S., and Gökmar, C.,** 2007: A CMOS Classifier Circuit using Neural Networks with Novel Architecture, *IEEE Transaction on Neural Networks*, Vol. 18, pp. 1845-1849.
- [55] **Yıldız, M., Minaei, S., and Gökmar, C.,** 2006: CMOS Realization of a Quantized-Output Classifier Circuit, *13th IEEE International Conference on Electronics, Circuits and Systems*, Nice, France, December 10-13, pp. 292-295.
- [56] **Morgül, A., and Temel, T.,** 2005: Current-mode level restoration circuit for multi-valued logic, *Electronics Letters*, Vol. 41, pp. 230-231.
- [57] **Hodges D., Jackson H., and Saleh R.,** 2004: *Analysis and Design of Digital Integrated Circuits*, Third edition, McGraw Hill, pp. 66-69, Newyork, USA.
- [58] **Yıldız, M., Minaei, S., and Gökmar, C.,** 2007: A Low-Power Multilevel-Output Classifier Circuit, *European Conference on Circuit Theory and Design*, Seville, Spain, August 26-30, pp. 747-750.

- [59] **Leong, P. H. W.**, 1993: Kakadu-A Low Power Analogue Neural Network Classifier, *International Journal of Neural Systems*, Vol. **4**, pp. 381-394.
- [60] **Lu, C., Shi, B., and Chen, L.**, 2002: A General Purpose Neural Network with on Chip BP Learning, *International Symposium on Circuits and System*, Arizona, USA, May 16-29, pp. 520-523.
- [61] **Gatt, E., and Micallef, J.**, 2002: Analogue Radial Basis Function Networks for Phoneme Recognition, *International Conference on Electronics, Circuits, and Systems*, Dubrovnik, Croatia, September 15-18, pp. 583-586.
- [62] **Zeng, H., Ying, H., and Xingbo, S.**, 2000: A fuzzy central cluster neural classifier, *Proceedings of the 3rd World Congress on Intelligent Control and Automation*, Hefei, China, June 28–July 2, pp. 1747-1750.
- [63] **Yıldız, M., Minaei, S., and Özoğuz, S.**, 2009: Linearly Weighted Classifier Circuit, *Northeast Workshop on Circuits and Systems*, konferans kitabında basılmak üzere kabul edilmiştir.
- [64] **Qian, D.**, 2007: Modified Fisher's Linear Discriminant Analysis for Hyperspectral Imagery, *IEEE Geoscience and Remote Sensing Letters*, Vol. **4**, pp. 503-507.
- [65] **Ferri, G., and Guerrini, N.**, 2001: High valued passive element simulation using low-voltage low- power current conveyor for fully integrated applications, *IEEE Transactions on Circuit and Systems II*, Vol. **48**, pp. 405-409
- [66] **Qi, L., and Donald, W. T.**, 1997: Principal Feature Classification, *IEEE Transaction on Neural Networks*, Vol. **8**, pp. 155-160.
- [67] **Jiang, G., Xiaoqing D., and Youshou W.**, 1999: On improvement of Multiple Discriminant Analysis Method for Discriminative Feature Extraction, *Systems, Man, and Cybernetics, Conference Proceedings*, October 12-15, pp. 915 – 920
- [68] **Çevikalp, H.**, 2006: Theoretical Analysis of Linear Discriminant Analysis Criteria, *IEEE 14th Signal Processing and Communications Applications*, Antalya, Turkey, April 17-19, pp. 1-4.
- [69] **Genç, İ., and Güzelış, C.**, 1998: Threshold Class CNNs with Input-Dependent Initial State, *IEEE International Workshop on Cellular Neural Networks and their Applications*, London, England, April 14-17, pp. 130-135
- [70] **Chou, C. H., and Chen, Y. C.**, 1989: A VLSI Architecture for real-time and flexible image template matching, *IEEE Trans. on Circuit and Systems*, Vol. **36**, pp. 1336-1342.

- [71] **Yıldız, M., Minaei, S., and Gökner, C.,** 2008: Realization and Template Matching Application of a CMOS Classifier Circuit, *IEEE Applied Electronics*, Pilsen, Czech Republic, September 10-11, pp. 231-234.
- [72] **Fisher, R.A.,** 1936: The use of multiple measurements in taxonomic problems, *Annual Eugenics*, Vol. **7**, pp. 179-188.
- [73] **Haberman, S.J.,** 1976: Generalized Residuals for Log-Linear Models, *Proceedings of the 9th International Biometrics Conference*, Boston, USA, pp. 104-122.

EKLER

EK A : AMS 0.35 μm MOSIS SPICE teknoloji parametreleri

EK B : Maksimum($\text{rank}(S_B)=c-1$)'in ispatı

EK C.1 : İris veri kümesi

EK C.2 : Haberman veri kümesi

EK A

Bu ekte benzetimlerde kullanılan NMOS ve PMOS tranzistorların 0.35 μm AMS CMOS SPICE teknoloji parametreleri verilmiştir.

NMOS:

(LEVEL=7 MOBMOD =1.000e+00 CAPMOD =2.000e+00 NLEV=0 K1=6.044e-01
K2=2.945e-03 K3=-1.72e K3B=6.325e-01 NCH=2.310e+17 VTH0=4.655e-01
VOFF=-5.72e-02 DVT0=2.227e+01 DVT1=1.051e DVT2=3.393e-03 KETA=-
6.21e-04 PSCBE1=2.756e+08 PSCBE2=9.645e-06 DVT0W=0.000 DVT1W=0.000
DVT2W=0.000 UA=1.000e-12 UB=1.723e-18 UC=5.756e-11 U0=4.035e+02
DSUB=5.000e-01 ETA0=3.085e-02 ETAB=-3.95e-02 NFACTOR=1.119e-01
EM=4.100e+07 PCLM=6.831e-01 PDIBLC1=1.076e-01 PDIBLC2=1.453e-03
DROUT=5.000e-01 A0=2.208e+00 A1=0.000 A2=1.000 PVAG=0.000
VSAT=1.178e+05 AGS=2.490e-01 B0=-1.76e-08 B1=0.000 DELTA=1.000e-02
PDIBLCB=2.583e-01 W0=1.184e-07 DLC=8.285e-09 DWC=2.676e-08
DWB=0.000 DWG=0.000 LL=0.000 LW=0.000 LWL=0.000 LLN=1.000
LWN=1.000e WL=0.000 WW=0.000 WWL=0.000 WLN=1.000e WWN=1.000
AT=3.300e+04 UTE=-1.80e KT1=-3.30e-01 KT2=2.200e-02 KT1L=0.000
UA1=0.000 UB1=0.000 UC1=0.000 PRT=0.000 CGDO=2.100e-10 CGSO=2.100e-
10 CGBO=1.100e-10 CGDL=0.000 CGSL=0.000 CKAPPA =6.000e-01 CF=0.000
ELM=5.000e XPART=1.000e CLC=1.000e-15 CLE=6.000e-01 RDSW=6.043e+02
CDSC=0.000 CDSCB=0.000 CDSCD=8.448e-05 PRWB=0.000 PRWG=0.000
CIT=1.000e-03 TOX=7.700e-09 NGATE=0.000e NLX=1.918e-07 ALPHA0=0.000
BETA0=3.000e+01 AF=1.400 KF=2.810e-27 EF=1.000 NOIA=1.000e+20
NOIB=5.000e+04 NOIC=-1.40e-12 LINT=-1.67e-08 WINT=2.676e-08 XJ=3.000e-
07 RSH=8.200e+01 JS=2.000e-05 CJ=9.300e-04 CJSW=2.800e-10 MJ=3.100e-01
MJSW=1.900e-01 PB=6.900e-01 TT=0.000e+00 PBSW=9.400e-01)

PMOS:

(LEVEL=7 MOBMOD =1.000e CAPMOD=2.000e NLEV=0 K1=5.675e-01 K2=-
4.39e-02 K3=4.540e K3B=-8.52e-01 NCH=1.032e+17 VTH0=-6.17e-01 VOFF=-

1.13e-01 DVT0=1.482e+00 DVT1=3.884e-01 DVT2=-1.15e-02 KETA=-2.56e-02
 PSCBE1=1.000e+09 PSCBE2=1.000e-08 DVT0W=0.000 DVT1W=0.000
 DVT2W=0.000 UA=2.120e-10 UB=8.290e-19 UC=-5.28e-11 U0=1.296e+02
 DSUB=5.000e-01 ETA0=2.293e-01 ETAB=-3.92e-03 NFACTOR=8.237e-01
 EM=4.100e+07 PCLM=2.979e+00 PDIBLC1=3.310e-02 PDIBLC2=1.000e-09
 DROUT =5.000e-01 A0=1.423e A1=0.000 A2=1.000 PVAG=0.000
 VSAT=2.000e+05 AGS=3.482e-01 B0=2.719e-07 B1=0.000 DELTA=1.000e-02
 PDIBLCB=-1.78e-02 W0=4.894e-08 DLC=-5.64e-08 DWC=3.845e-08 DWB=0.000
 DWG=0.000 LL=0.000 LW=0.000 LWL=0.000 LLN=1.000 LWN=1.000
 WL=0.000 WW=0.000 WWL=0.000 WLN=1.000 WWN=1.000 AT=3.300e+04
 UTE=-1.35e+00 KT1=-5.70e-01 KT2=2.200e-02 KT1L=0.000 UA1=0.000
 UB1=0.000 UC1=0.000 PRT=0.000 CGDO=2.100e-10 CGSO=2.100e-10
 CGBO=1.100e-10 CGDL=0.000 CGSL=0.000 CKAPPA =6.000e-01 CF=0.000
 ELM=5.000 XPART=1.000 CLC=1.000e-15 CLE=6.000e-01 RDSW=1.853e+03
 CDSC=6.994e-04 CDSCB=2.943e-04 CDSCD=1.970e-04 PRWB=0.000
 PRWG=0.000 CIT=1.173e-04 TOX=7.700e-09 NGATE=0.000 NLX=1.770e-07
 ALPHA0=0.000 BETA0=3.000e+01 AF=1.290e KF=1.090e-27 EF=1.000e
 NOIA=1.000e+20 NOIB=5.000e+04 NOIC=-1.40e-12 LINT=-8.14e-08
 WINT=3.845e-08 XJ=3.000e-07 RSH=1.560e+02 JS=2.000e-05 CJ=1.420e-03
 CJSW=3.800e-10 MJ=5.500e-01 MJSW=3.900e-01 PB=1.020e+00 TT=0.000e
 PBSW=9.400e-01)

EK B

Bu ekte, maksimum(rank(S_B))= $c-1$ ’ olduğu ispatlanmıştır.

$$\begin{aligned}
 S_B = & n_1 \left(\bar{\mu}_1 - \frac{1}{c}(\bar{\mu}_1 + \bar{\mu}_2 + \dots + \bar{\mu}_c) \right) \left(\bar{\mu}_1 - \frac{1}{c}(\bar{\mu}_1 + \bar{\mu}_2 + \dots + \bar{\mu}_c) \right)^T + \\
 & n_2 \left(\bar{\mu}_2 - \frac{1}{c}(\bar{\mu}_1 + \bar{\mu}_2 + \dots + \bar{\mu}_c) \right) \left(\bar{\mu}_2 - \frac{1}{c}(\bar{\mu}_1 + \bar{\mu}_2 + \dots + \bar{\mu}_c) \right)^T + \dots + \\
 & n_c \left(\bar{\mu}_c - \frac{1}{c}(\bar{\mu}_1 + \bar{\mu}_2 + \dots + \bar{\mu}_c) \right) \left(\bar{\mu}_c - \frac{1}{c}(\bar{\mu}_1 + \bar{\mu}_2 + \dots + \bar{\mu}_c) \right)^T
 \end{aligned} \tag{B.1}$$

$$\begin{aligned}
 S_B = & \frac{n_1}{c^2} ((c-1)\bar{\mu}_1 - \bar{\mu}_2 - \dots - \bar{\mu}_c) ((c-1)\bar{\mu}_1 - \bar{\mu}_2 - \dots - \bar{\mu}_c)^T + \\
 & \frac{n_2}{c^2} (-\bar{\mu}_1 + (c-1)\bar{\mu}_2 - \dots - \bar{\mu}_c) (-\bar{\mu}_1 + (c-1)\bar{\mu}_2 - \dots - \bar{\mu}_c)^T + \dots + \\
 & \frac{n_c}{c^2} (-\bar{\mu}_1 - \bar{\mu}_2 - \dots + (c-1)\bar{\mu}_c) (-\bar{\mu}_1 - \bar{\mu}_2 - \dots + (c-1)\bar{\mu}_c)^T
 \end{aligned} \tag{B.2}$$

S_B matrisinin ifadesinde $\mathfrak{S} = [\bar{\mu}_1 \ \bar{\mu}_2 \dots \bar{\mu}_c]$ tanımı kullanılırsa, toplam üç matrisin çarpımı biçiminde aşağıda gösterildiği şekilde ifade edilebilir.

$$S_B = \mathfrak{S} \left\{ \begin{array}{l} \frac{n_1}{c^2} \begin{bmatrix} c-1 \\ -1 \\ \cdot \\ -1 \end{bmatrix} [c-1 \ -1 \ \cdot \ \cdot \ -1] + \frac{n_2}{c^2} \begin{bmatrix} -1 \\ c-1 \\ -1 \\ \cdot \\ -1 \end{bmatrix} [-1 \ c-1 \ -1 \ \cdot \ -1] + \dots + \\ \frac{n_c}{c^2} \begin{bmatrix} -1 \\ \cdot \\ \cdot \\ -1 \\ c-1 \end{bmatrix} [-1 \ \cdot \ \cdot \ -1 \ c-1] \end{array} \right\} \mathfrak{S}^T \tag{B.3}$$

(B.3) ifadesinde ortadaki toplamda yer alan matrislerin hepsinin satırları aynı lineer bağımlılığı (satırların toplamaları sıfır ediyor) sağladığından toplam matrisin dolayısıyla da S_B matrisinin rankı $c-1$ sayısından büyük olamaz.

EK C.1

Çizelge C.1 : İris veri kümesi.

Çanak Yapağı Uzunluğu [cm]	Çanak Yapağı Genişliği [cm]	Taç Yapağı Uzunluğu [cm]	Taç Yapağı Genişliği [cm]	Sınıf
4.30	2.30	1.40	0.20	Setosa
4.40	2.90	1.40	0.20	Setosa
4.40	3.00	1.30	0.20	Setosa
4.40	3.00	1.50	0.20	Setosa
4.50	3.00	1.40	0.20	Setosa
4.60	3.00	1.70	0.40	Setosa
4.60	3.00	1.40	0.30	Setosa
4.60	3.00	1.50	0.20	Setosa
4.60	3.10	1.40	0.20	Setosa
4.70	3.10	1.50	0.10	Setosa
4.70	3.10	1.50	0.20	Setosa
4.80	3.10	1.60	0.20	Setosa
4.80	3.10	1.40	0.10	Setosa
4.80	3.20	1.10	0.10	Setosa
4.80	3.20	1.20	0.20	Setosa
4.80	3.20	1.50	0.40	Setosa
4.90	3.20	1.30	0.40	Setosa
4.90	3.20	1.40	0.30	Setosa
4.90	3.30	1.70	0.30	Setosa
4.90	3.30	1.50	0.30	Setosa
5.00	3.40	1.70	0.20	Setosa
5.00	3.40	1.50	0.40	Setosa
5.00	3.40	1.00	0.20	Setosa
5.00	3.40	1.70	0.50	Setosa
5.00	3.40	1.90	0.20	Setosa
5.00	3.40	1.60	0.20	Setosa
5.00	3.40	1.60	0.40	Setosa
5.00	3.40	1.50	0.20	Setosa
5.10	3.40	1.40	0.20	Setosa
5.10	3.50	1.60	0.20	Setosa
5.10	3.50	1.60	0.20	Setosa
5.10	3.50	1.50	0.40	Setosa
5.10	3.50	1.50	0.10	Setosa
5.10	3.50	1.40	0.20	Setosa
5.10	3.50	1.50	0.10	Setosa
5.10	3.60	1.20	0.20	Setosa
5.20	3.60	1.30	0.20	Setosa
5.20	3.70	1.50	0.10	Setosa
5.20	3.70	1.30	0.20	Setosa
5.30	3.70	1.50	0.20	Setosa
5.40	3.80	1.30	0.30	Setosa
5.40	3.80	1.30	0.30	Setosa
5.40	3.80	1.30	0.20	Setosa
5.40	3.80	1.60	0.60	Setosa
5.40	3.90	1.90	0.40	Setosa
5.50	3.90	1.40	0.30	Setosa
5.50	4.00	1.60	0.20	Setosa
5.70	4.10	1.40	0.20	Setosa
5.70	4.20	1.50	0.20	Setosa

Çizelge C.1 : İris veri kümesi (devam).

Çanak Yaprığı Uzunluğu [cm]	Çanak Yaprığı Genişliği [cm]	Taç Yaprığı Uzunluğu [cm]	Taç Yaprığı Genişliği [cm]	Sınıf
5.80	4.40	1.40	0.20	Setosa
4.90	2.00	4.70	1.40	Versicolour
5.00	2.20	4.50	1.50	Versicolour
5.00	2.20	4.90	1.50	Versicolour
5.10	2.30	4.00	1.30	Versicolour
5.20	2.30	4.60	1.50	Versicolour
5.40	2.30	4.50	1.30	Versicolour
5.50	2.40	4.70	1.60	Versicolour
5.50	2.40	3.30	1.00	Versicolour
5.50	2.40	4.60	1.30	Versicolour
5.50	2.50	3.90	1.40	Versicolour
5.50	2.50	3.50	1.00	Versicolour
5.60	2.50	4.20	1.50	Versicolour
5.60	2.50	4.00	1.00	Versicolour
5.60	2.60	4.70	1.40	Versicolour
5.60	2.60	3.60	1.30	Versicolour
5.60	2.60	4.40	1.40	Versicolour
5.70	2.70	4.50	1.50	Versicolour
5.70	2.70	4.10	1.00	Versicolour
5.70	2.70	4.50	1.50	Versicolour
5.70	2.70	3.90	1.10	Versicolour
5.70	2.70	4.80	1.80	Versicolour
5.80	2.80	4.00	1.30	Versicolour
5.80	2.80	4.90	1.50	Versicolour
5.80	2.80	4.70	1.20	Versicolour
5.90	2.80	4.30	1.30	Versicolour
5.90	2.80	4.40	1.40	Versicolour
6.00	2.80	4.80	1.40	Versicolour
6.00	2.90	5.00	1.70	Versicolour
6.00	2.90	4.50	1.50	Versicolour
6.00	2.90	3.50	1.00	Versicolour
6.10	2.90	3.80	1.10	Versicolour
6.10	2.90	3.70	1.00	Versicolour
6.10	2.90	3.90	1.20	Versicolour
6.10	2.90	5.10	1.60	Versicolour
6.20	3.00	4.50	1.50	Versicolour
6.20	3.00	4.50	1.60	Versicolour
6.30	3.00	4.70	1.50	Versicolour
6.30	3.00	4.40	1.30	Versicolour
6.30	3.00	4.10	1.30	Versicolour
6.40	3.00	4.00	1.30	Versicolour
6.40	3.00	4.40	1.20	Versicolour
6.50	3.00	4.60	1.40	Versicolour
6.60	3.10	4.00	1.20	Versicolour
6.60	3.10	3.30	1.00	Versicolour
6.70	3.10	4.20	1.30	Versicolour
6.70	3.20	4.20	1.20	Versicolour
6.70	3.20	4.20	1.30	Versicolour
6.80	3.20	4.30	1.30	Versicolour
6.90	3.30	3.00	1.10	Versicolour
7.00	3.40	4.10	1.30	Versicolour
4.90	2.20	6.00	2.50	Virginica

Çizelge C.1 : İris veri kümesi (devam).

Çanak Yapağı Uzunluğu [cm]	Çanak Yapağı Genişliği [cm]	Taç Yapağı Uzunluğu [cm]	Taç Yapağı Genişliği [cm]	Sınıf
5.60	2.50	5.10	1.90	Virginica
5.70	2.50	5.90	2.10	Virginica
5.80	2.50	5.60	1.80	Virginica
5.80	2.50	5.80	2.20	Virginica
5.80	2.60	6.60	2.10	Virginica
5.90	2.60	4.50	1.70	Virginica
6.00	2.70	6.30	1.80	Virginica
6.00	2.70	5.80	1.80	Virginica
6.10	2.70	6.10	2.50	Virginica
6.10	2.70	5.10	2.00	Virginica
6.20	2.80	5.30	1.90	Virginica
6.20	2.80	5.50	2.10	Virginica
6.30	2.80	5.00	2.00	Virginica
6.30	2.80	5.10	2.40	Virginica
6.30	2.80	5.30	2.30	Virginica
6.30	2.80	5.50	1.80	Virginica
6.30	2.80	6.70	2.20	Virginica
6.30	2.80	6.90	2.30	Virginica
6.40	2.90	5.00	1.50	Virginica
6.40	2.90	5.70	2.30	Virginica
6.40	3.00	4.90	2.00	Virginica
6.40	3.00	6.70	2.00	Virginica
6.40	3.00	4.90	1.80	Virginica
6.50	3.00	5.70	2.10	Virginica
6.50	3.00	6.00	1.80	Virginica
6.50	3.00	4.80	1.80	Virginica
6.50	3.00	4.90	1.80	Virginica
6.70	3.00	5.60	2.10	Virginica
6.70	3.00	5.80	1.60	Virginica
6.70	3.00	6.10	1.90	Virginica
6.70	3.00	6.40	2.00	Virginica
6.70	3.00	5.60	2.20	Virginica
6.80	3.10	5.10	1.50	Virginica
6.80	3.10	5.60	1.40	Virginica
6.90	3.10	6.10	2.30	Virginica
6.90	3.10	5.60	2.40	Virginica
6.90	3.20	5.50	1.80	Virginica
7.10	3.20	4.80	1.80	Virginica
7.20	3.20	5.40	2.10	Virginica
7.20	3.20	5.60	2.40	Virginica
7.20	3.20	5.10	2.30	Virginica
7.30	3.30	5.10	1.90	Virginica
7.40	3.30	5.90	2.30	Virginica
7.60	3.30	5.70	2.50	Virginica
7.70	3.40	5.20	2.30	Virginica
7.70	3.40	5.00	1.90	Virginica
7.70	3.60	5.20	2.00	Virginica
7.70	3.80	5.40	2.30	Virginica
7.90	3.80	5.10	1.80	Virginica

EK C.2

Bu ekte veri sınıflandırıcılarının uygulamalarında eğitim ve test amacı ile kullanılan Haberman verisi verilmiştir. x_1 hastanın yaşını, x_2 ameliyat tarihini ve x_3 pozitif netice veren göğüs kanserli hasta sayısını göstermektedir. c_1 beş yıldan daha uzun yaşayan, c_2 beş yıldan daha az yaşayan hasta sınıfını göstermektedir.

Çizelge C.2 : Haberman veri kümesi.

x_1	x_2	x_3	Sınıf	x_1	x_2	x_3	Sınıf	x_1	x_2	x_3	Sınıf	x_1	x_2	x_3	Sınıf
30	64	1	c_1	37	63	0	c_1	39	59	2	c_1	42	62	20	c_1
30	62	3	c_1	37	58	0	c_1	39	63	4	c_1	42	65	0	c_1
30	65	0	c_1	37	59	6	c_1	40	58	2	c_1	42	63	1	c_1
31	59	2	c_1	37	60	15	c_1	40	58	0	c_1	43	63	14	c_1
31	65	4	c_1	37	63	0	c_1	40	65	0	c_1	43	64	2	c_1
33	58	10	c_1	38	59	2	c_1	41	58	0	c_1	43	64	3	c_1
33	60	0	c_1	38	60	0	c_1	41	59	8	c_1	43	60	0	c_1
34	60	1	c_1	38	62	3	c_1	41	59	0	c_1	43	65	0	c_1
34	61	10	c_1	38	64	1	c_1	41	64	0	c_1	43	66	4	c_1
34	67	7	c_1	38	66	0	c_1	41	69	8	c_1	44	61	0	c_1
34	60	0	c_1	38	66	11	c_1	41	65	0	c_1	44	63	1	c_1
35	64	13	c_1	38	60	1	c_1	41	65	0	c_1	44	61	0	c_1
35	63	0	c_1	38	67	5	c_1	42	58	0	c_1	44	67	16	c_1
36	60	1	c_1	39	63	0	c_1	42	60	1	c_1	45	60	0	c_1
36	69	0	c_1	39	67	0	c_1	42	59	2	c_1	45	67	0	c_1
45	59	14	c_1	49	62	0	c_1	52	61	0	c_1	54	69	7	c_1
45	64	0	c_1	49	66	0	c_1	52	63	4	c_1	54	63	19	c_1
45	68	0	c_1	49	60	1	c_1	52	69	0	c_1	54	58	1	c_1
45	67	1	c_1	49	62	1	c_1	52	60	4	c_1	54	62	0	c_1
46	62	0	c_1	49	63	3	c_1	52	60	5	c_1	55	58	1	c_1
46	58	3	c_1	49	61	0	c_1	52	62	0	c_1	55	58	0	c_1
46	63	0	c_1	49	67	1	c_1	52	62	1	c_1	55	58	1	c_1
47	61	0	c_1	50	59	0	c_1	52	64	0	c_1	55	66	18	c_1
47	63	6	c_1	50	61	6	c_1	52	65	0	c_1	55	66	0	c_1
47	66	0	c_1	50	61	0	c_1	52	68	0	c_1	55	69	3	c_1
47	67	0	c_1	50	63	1	c_1	53	58	1	c_1	55	69	22	c_1
47	58	3	c_1	50	58	1	c_1	53	60	1	c_1	55	67	1	c_1
47	60	4	c_1	50	59	2	c_1	53	60	2	c_1	56	60	0	c_1
47	68	4	c_1	50	61	0	c_1	53	61	1	c_1	56	66	2	c_1
47	66	12	c_1	50	64	0	c_1	53	63	0	c_1	56	66	1	c_1
48	61	8	c_1	50	65	4	c_1	54	59	7	c_1	56	67	0	c_1

Çizelge C.2 : Haberman veri kümesi (devam).

x_1	x_2	x_3	Sınıf	x_1	x_2	x_3	Sınıf	x_1	x_2	x_3	Sınıf	x_1	x_2	x_3	Sınıf
58	60	3	c_1	59	67	3	c_1	61	64	0	c_1	76	67	0	c_1
58	61	1	c_1	60	61	1	c_1	61	65	8	c_1	77	65	3	c_1
58	67	0	c_1	59	64	7	c_1	61	68	0	c_1	34	59	0	c_2
58	58	0	c_1	63	63	0	c_1	61	59	0	c_1	34	66	9	c_2
58	58	3	c_1	63	66	0	c_1	62	62	6	c_1	38	69	21	c_2
58	61	2	c_1	63	61	9	c_1	62	66	0	c_1	39	66	0	c_2
59	60	0	c_1	63	61	28	c_1	62	66	0	c_1	41	60	23	c_2
65	58	0	c_1	64	58	0	c_1	62	58	0	c_1	41	64	0	c_2
65	64	0	c_1	64	65	22	c_1	63	61	0	c_1	41	67	0	c_2
65	67	0	c_1	64	66	0	c_1	63	62	0	c_1	42	69	1	c_2
65	59	2	c_1	64	61	0	c_1	63	63	0	c_1	42	59	0	c_2
65	64	0	c_1	64	68	0	c_1	67	66	0	c_1	43	58	52	c_2
71	68	2	c_1	69	65	0	c_1	67	61	0	c_1	43	59	2	c_2
70	63	0	c_1	70	66	14	c_1	68	67	0	c_1	43	64	0	c_2
72	58	0	c_1	70	67	0	c_1	68	68	0	c_1	43	64	0	c_2
72	64	0	c_1	70	68	0	c_1	69	60	0	c_1	44	64	6	c_2
72	67	3	c_1	65	67	1	c_1	74	63	0	c_1	44	58	9	c_2
73	62	0	c_1	73	68	0	c_1	75	62	1	c_1	44	63	19	c_2
46	62	5	c_2	48	67	7	c_2	52	59	2	c_2	45	65	6	c_2
46	65	20	c_2	49	63	0	c_2	52	62	3	c_2	45	66	0	c_2
47	63	23	c_2	49	64	10	c_2	52	66	4	c_2	45	67	1	c_2
47	62	0	c_2	50	63	13	c_2	53	58	4	c_2	46	58	2	c_2
47	65	0	c_2	50	64	0	c_2	53	65	1	c_2	46	69	3	c_2
48	58	11	c_2	51	59	13	c_2	53	59	3	c_2	56	65	9	c_2
48	58	11	c_2	51	59	3	c_2	53	60	9	c_2	56	66	3	c_2
54	65	5	c_2	52	69	3	c_2	53	63	24	c_2	57	61	5	c_2
54	68	7	c_2	56	65	9	c_2	53	65	12	c_2	57	62	14	c_2
55	63	6	c_2	56	66	3	c_2	54	60	11	c_2	57	64	1	c_2
55	68	15	c_2	57	61	5	c_2	54	65	23	c_2	59	62	35	c_2
62	59	13	c_2	57	64	1	c_2	59	62	35	c_2	60	65	0	c_2
62	58	0	c_2	61	62	5	c_2	60	59	17	c_2	61	62	5	c_2
62	65	19	c_2	61	65	0	c_2	60	65	0	c_2	61	65	0	c_2
63	60	1	c_2	65	66	15	c_2	67	63	1	c_2	70	58	0	c_2
65	58	0	c_2	66	58	0	c_2	69	67	8	c_2	70	58	4	c_2
65	61	2	c_2	66	61	13	c_2	74	65	3	c_2	72	63	0	c_2
65	62	22	c_2	67	64	8	c_2	78	65	1	c_2	83	58	2	c_2
57	63	0	c_1	59	63	0	c_1	60	67	2	c_1	66	58	0	c_1
57	64	0	c_1	59	64	1	c_1	60	61	25	c_1	66	58	1	c_1
57	67	0	c_1	59	64	0	c_1	61	59	0	c_1	67	66	0	c_1

ÖZGEÇMİŞ



Ad Soyad: Merih YILDIZ

Doğum Yeri ve Tarihi: Zonguldak, 11-08-1978

Adres: Acarlar Sitesi C-4 Blok d:11 İstanbul.

Lisans Üniversitesi: İstanbul Teknik Üniversitesi, Elektronik ve Haberleşme Mühendisliği Bölümü, 2000.

Yüksek Lisans Üniversitesi: İstanbul Teknik Üniversitesi, Elektronik ve Haberleşme Mühendisliği Bölümü, 2003.

Yayın Listesi:

- **Yıldız, M.**, Minaei, S., and Özoğuz, S., 2009: Linearly Weighted Classifier Circuit, *Northeast Workshop on Circuits and Systems*, konferans kitabında basılmak üzere kabul edilmiştir.
- **Yıldız, M.**, Minaei, S., and Gökner, C., 2008: Realization and Template Matching Application of a CMOS Classifier Circuit, *IEEE Applied Electronics*, Pilsen, Czech Republic, September 10-11, pp. 231-234.
- **Yıldız, M.**, Minaei, S., and Gökner, C., 2007: A CMOS Classifier Circuit using Neural Networks with Novel Architecture, *IEEE Transaction on Neural Networks*, Vol. **18**, pp. 1845-1849.
- **Yıldız, M.**, Minaei, S., and Gökner, C., 2007: A Low-Power Multilevel-Output Classifier Circuit, *European Conference on Circuit Theory and Design*, August 26-30 Ağustos, Seville, Spain, pp. 747-750.
- **Yıldız, M.**, Minaei, S., and Gökner, C., 2006: Current Mode Double Threshold Neuron Activation Function, *Complex Computing Networks, Springer in Physics Series*, Vol. **104**, pp. 267-274.

- Minaei S., **Yıldız, M.**, Türköz, S., and Kuntman, H., 2003: High Swing CMOS Realization for Third Generation Current Conveyor (CCIII), *Istanbul University Journal of Electrical and Electronics*, Vol **3**, pp. 819-826.
- **Yıldız, M.**, Minaei, S., and Gökner, C., 2006: CMOS Realization of a Quantized-Output Classifier Circuit, *13th IEEE International Conference on Electronics, Circuits and Systems*, December 10-13, Nice, France, pp. 292-295.
- **Yıldız, M.**, Minaei, S., and Gökner, C., 2005: Current Mode Double Threshold Neuron Activation Function, *Complex Computing-Networks: Brain-like and Wave-oriented Electrodynamical Algorithms Springer Proceedings in Physics*, Vol. **104**, pp. 267-274.
- Minaei, S., **Yıldız, M.**, and Kuntman, H., 2004: New Realization of Floating Lossless Inductance and R-L Impedance Simulators, *Northeast Workshop on Circuits and Systems*, June 20-23, Montreal, Canada, pp. 313-316.
- Minaei, S., **Yıldız, M.**, and Kuntman, H., 2004: New Realization of Voltage-Mode Multifunction Filters without External Passive Elements, *2th IEEE Mediterranean Electrotechnical Conference*, Dubrovnik, Croatia, May 12-15, pp. 99-102.
- Minaei S., **Yıldız, M.**, Metin, B., and Çicekoğlu, O., 2004: İki Ucu Serbest Kayıpsız Endüktans ve R-L Empedans Benzetimlerinin Aktif Devreler İle Gerçeklenmesi, *Elektrik-Elektronik ve Bilgisayar Mühendisliği Sempozyumu*, December 8-12, Bursa, Turkey, pp. 96-99.
- Minaei, S., **Yıldız, M.**, and Kuntman, H., 2004: Sadece Aktif Eleman Kullanılarak Süzgeç Tasarımı ve Gerçekleştirilmesi, *IEEE 12. Sinyal İşleme ve İletişim Uygulamaları Kurultayı*, Kuşadası, Turkey, April 28-30, pp. 630-633.
- Minaei, S., **Yıldız, M.**, Kuntman, H., and Turkoz, S., 2002: Yeni Yüksek Başarımli CMOS Üçüncü Kuşak Akım Taşıyıcı (CCIII), *Elektrik-Elektronik ve Bilgisayar Mühendisliği Sempozyumu*, Bursa, Turkey, December 18-22, pp. 108-112.
- Minaei, S., **Yıldız, M.**, Kuntman, H., and Turkoz, S., 2002: High Performance CMOS Realization of The Third Generation Current conveyor (CCIIIs), *45th IEEE International Midwest Symposium on Circuit and Systems*, August 4-7, Oklahoma, USA, pp. 307-310.

